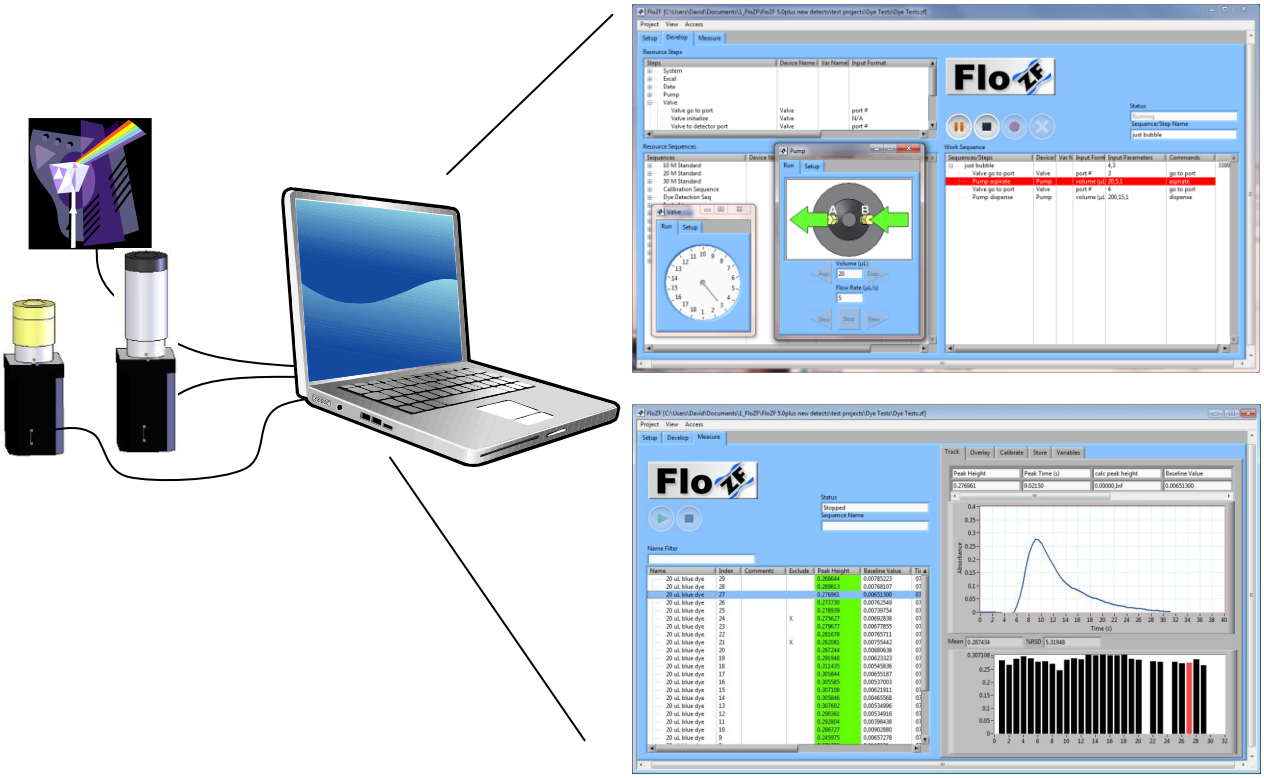
FloZF 5.0 User Manual



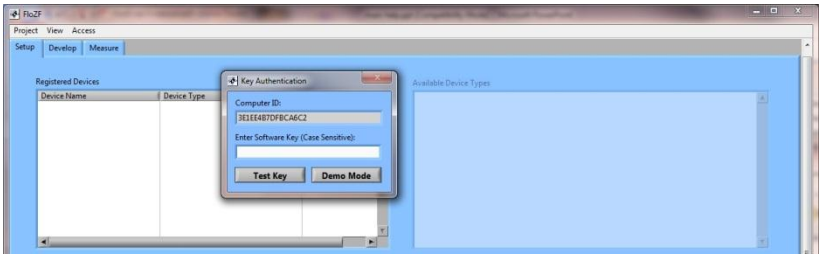
For easy navigation, view the PDF bookmarks

1 General Operation

1.1 Tutorial

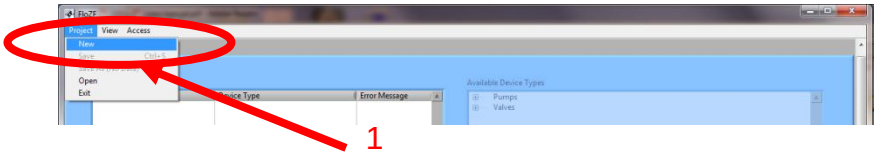
1.1.1 First Time Startup

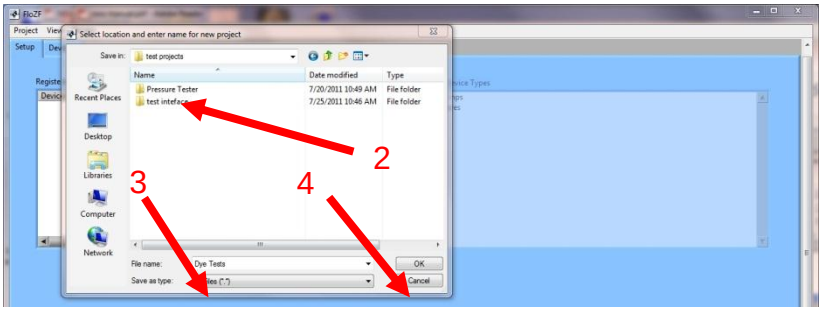
Select Demo Mode to allow operation of milliGAT (SmoothFlow) pumps, Valco valves, and Nymotion plate sampler or if you have a Software Key, enter it now and select Test Key.



1.1.2 Creating a New Project

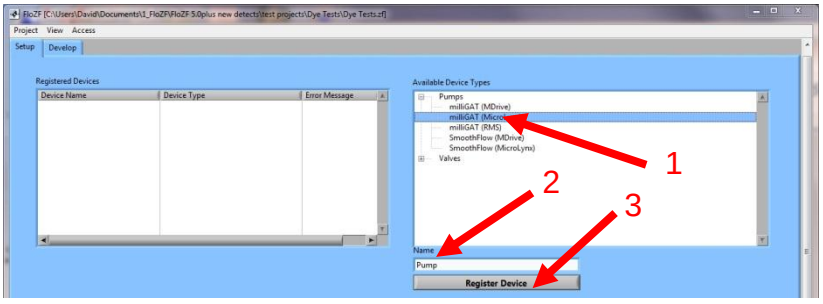
1) Select New under the Project menu
2) Navigate to the location for the project
3) Enter a descriptive name for the project such as “Ammonia”
4) Select OK. A folder with the project name is created to store all project files



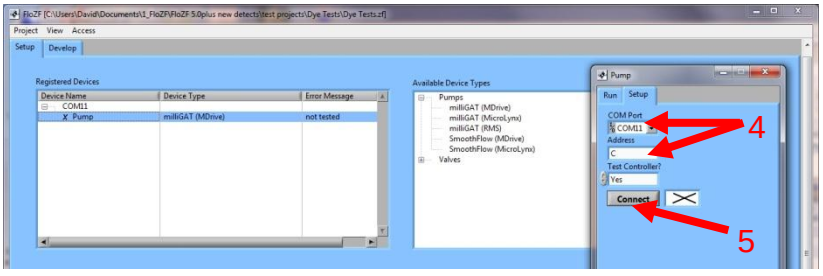


1.1.3 Adding Devices to a Project

1) Select the device type from the corresponding category
2) Enter a Name for the device in the field that appears below
3) Select Register Device. An instrument panel window will pop up.

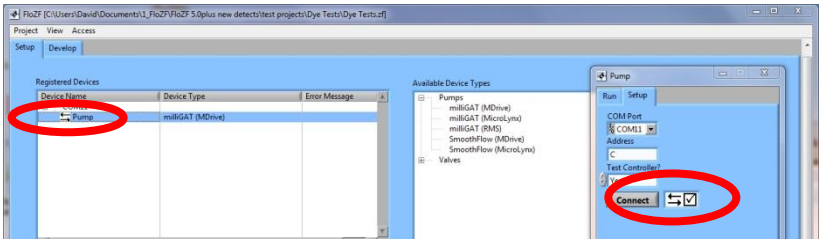


4) Enter communications settings on the instrument panel Setup tab such as the COM Port (by default there is no Address for a milliGAT



pump with MicroLynx controller so leave the Address field blank)

5) Select Connect
6) If a connection is made, the indicator changes. Otherwise, the communications settings may need to be changed

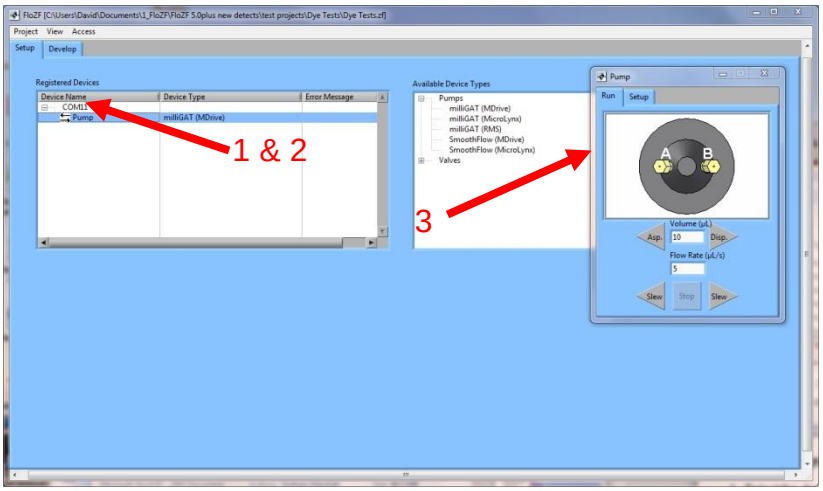


1.1.4 Accessing & Deleting Devices

1) To delete a device, click the device name once in the Registered Devices panel so that it is highlighted and press Control-Delete

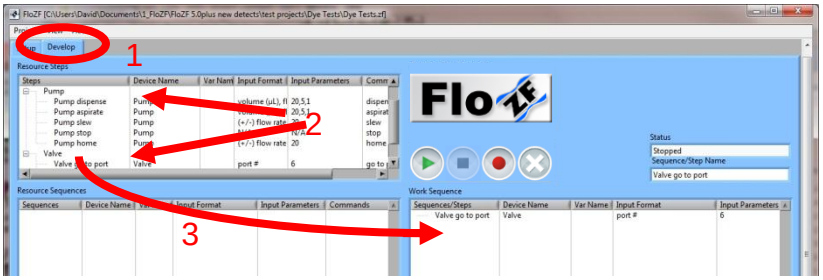
2) To re-open a device instrument panel, click the name in the Registered Devices panel twice (once if it is already highlighted)

3) The device can then be operated "manually" from the Run page of the instrument panel



1.1.5 Modifying and Executing Steps

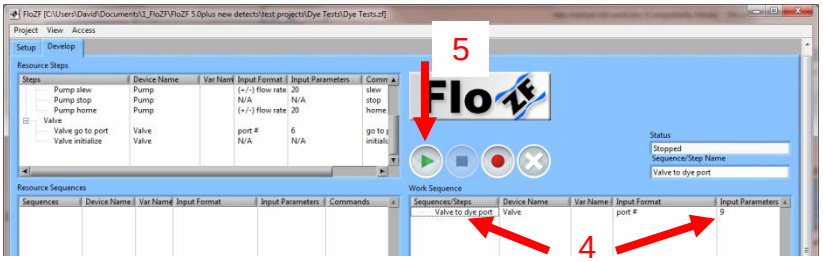
1) Select the **Develop** page



2) Every registered device has default commands, “steps”, that are accessed from the Resource Steps panel

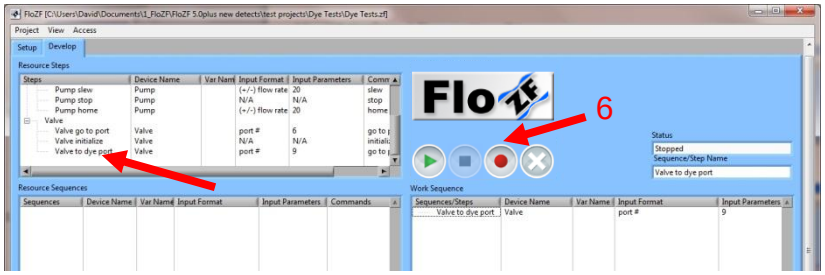
3) To modify or execute a step, drag it into the Work Sequence panel

4) The name and parameters of a step can be modified in the first and fourth columns by selecting on the text with two slow clicks



5) The step can be executed by pressing the play button

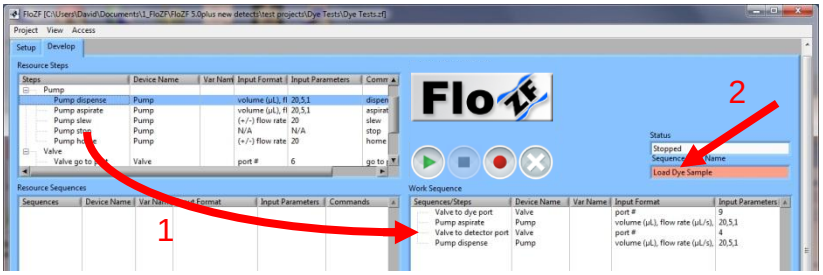
6) To save the modified step, press the Rec button. This adds the step to the Resource Steps panel.



7) To delete a step, select it and press Control-Delete

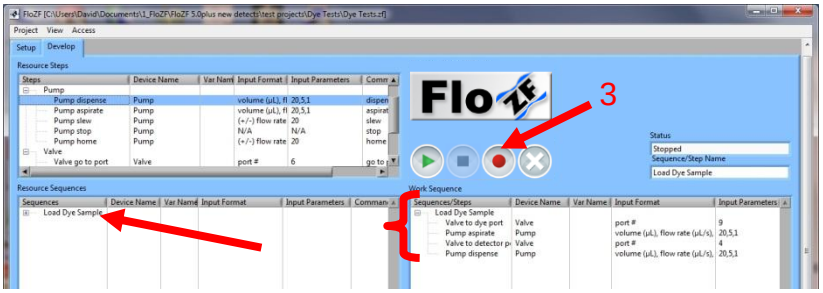
1.1.6 Building Sequences

1) To build a sequence from steps, drag two or more steps into the Work Sequence panel



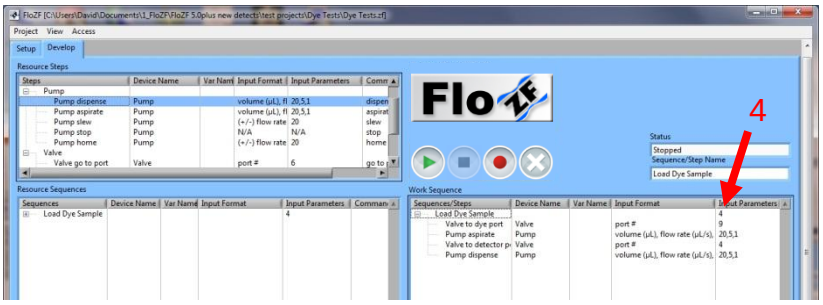
2) Enter a name for the sequence into the Sequence/Step Name field

3) Press the Rec button. The steps are grouped together under the name of the sequence and the sequence appears in the Resource Sequences panel



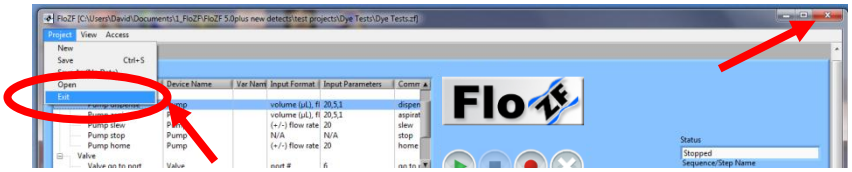
4) If a sequence is to be repeated, enter the number of times it is to be repeated in the Parameters column

5) Sequences can also be dragged from the Resource Sequences panel to the Work Sequence panel to build new sequences



1.1.7 Exiting FloZF

FloZF can be exited by either selecting Exit under the Project menu or by closing out the window.



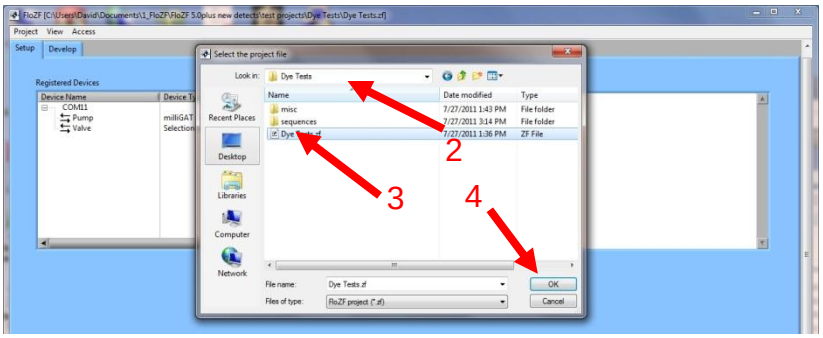
1.1.8 Loading an Existing Project from FloZF

1) Select Open under the Project menu

2) Navigate into the project folder that was created by FloZF

3) Select the file having the project name

4) Press OK



1.2 Project Management

FloZF **Projects** contain

- Device configuration information,
- Sequences to control the devices, and
- Data acquired from device detectors.

1.2.1 File Structure

All of the files for a **Project** are located in one folder with the name of the Project. The top level of the folder hierarchy is as follows.

- project name
  - project_name.zf (the **project file)**
  - sequences
  - misc
    - resource steps.xml
    - devices.xml
    - sequence log.sqlg
  - data folders
    - first_data_folder_name
    - …

The purpose of these files and folders are made clear in their corresponding sections.

1.2.2 Loading a Project from Windows

If FloZF is closed, a project can be opened with FloZF directly by double clicking on the project file.

1.2.3 Creating a Shortcut to the Project File

A shortcut to a project file can be created in a convenient place such as the desktop by right-clicking on the project file and selecting Create shortcut. The shortcut that is created can then be dragged to the desktop. Double-clicking on the shortcut will open the project in FloZF if FloZF is closed.

1.2.4 Saving a Project

Changes to device configurations are saved by selecting **Project** then **Save** in the menu at the top of the main window. All other information is saved at appropriate times as changes are made and when the software is exited.

1.2.5 Copying a Project from FloZF

1) Select **Project** then **Save As** in the menu at the top of the main window
2) Navigate to the Project folder
3) In the window that appears, navigate to the directory in which you want to create the new Project.
4) Enter a name for the new Project.
5) Select OK

1.2.6 Password Protecting a Project

To lockout users from having access to the Development page for the current project, select Access, Create/Modify Password from the top menu. In the window that appears, enter a password and select OK. Then, select Access, Lockout Access and the Development page will be hidden.

1.2.7 Unlocking a Password Projected Project

To regain access to the Development page for a password protected project, select Access, Enter Password from the top window. In the window that appears, type in the password and select OK.

1.2.8 Corrupt Projects and Software Updates

Should the project files somehow become corrupt or if the project is not compatible with a newer version of the software, the resource Steps and Sequences from the existing Project can be used in a new Project. Data files from the existing Project can be viewed as text documents using third-party software packages like Notepad or Excel. Device configuration information must be re-entered.

To build a new project with the existing Sequences:

1) Create a new Project
2) Re-create the device list using the same names as used before for each device. If you use different device names, the Sequences will no longer work.
3) Exit FloZF
4) Copy the old Sequences into the new Sequences folder
5) Load the new Project

1.3 Devices

1.3.1 Connection States

A connection with a device can be established, after filling in the communications settings fields, by pressing the **Connect** button. Two-way communication with the device is automatically tested unless the option to skip this testing is available for the device and has been selected by the user. Most devices require two-way communications to operate and display in the event of unsuccessful testing and for successful testing. Some non-detector serial devices provide the option for whether or not two-way communications with the device controller should be tested. For these devices, the symbol is displayed when the option to not test communication with the controller is selected, so long as the COM port can be opened. The communication status symbols are also displayed next to the device name in the Registered Devices list. If proper communication is not established with a device, check your settings and electrical connections.

1.3.2 Instrument View

An organized view of desired instrument panels can be created by opening just the desired panels, locating them in their desired positions, and selecting View, Record Instrument View from the top menu. This view can then be displayed or hid by selecting Display Instrument View or Hide Instrument View respectively.

1.3.3 COM Port Structure

A serial device is initially grouped in the registered devices panel under the heading XCOM when it is added to the project. When the COM port is changed on an instrument panel, the port is opened automatically and the device name is grouped under the COM port name heading in the registered devices panel.

1.3.4 Reconnecting to a COM Port

A closed or faulty COM port can be reopened by selecting the COM port name in the Registered Devices panel and selecting the Connect button that appears below the panel.

1.3.5 Changing Group COM Ports or COM Port Settings

The COM port or the COM port settings for a group of devices can be changed by selecting the COM port name in the registered devices panel, changing the COM port or settings in the panel that appears below, and selecting Connect. The COM port settings (Baud Rate, Data Bits, etc…) can only be changed if all of the devices on the COM port are General serial devices or General serial detectors.

1.3.6 Changing Device COM Ports

Single device COM ports can only be changed from the devices Instrument Panel Setup page.

1.4 Sequences

1.4.1 Sequence Flow Control

Steps and subsequences within a sequence are issued in sequential order from top to bottom. Most steps are managed by the sequence controller such that the intended operation of the step is completed before moving onto the next step in the sequence. However, some steps such as the detector "start acquire" step simply initiate an action and then immediately move on to the next step in the sequence. For this type of step there is typically an accompanying "stop" step that is called later on in a sequence to complete the intended operation.

1.4.2 System Steps (Excluding Variable Steps)

System commands/steps are available once the first device is added to a project.

prompt user

Parameters: message to user

Description: Launches a window that displays the **message to user** text followed by a button labeled OK. Execution of the Sequence is paused until the user pushes the button. This Step offers a convenient means of prompting the user to perform a specific action before proceeding, e.g. Check Carrier reservoir.

wait

Parameters: times(s)

Description: Pauses the Sequence for the specified **time**. If this time is greater than 30 seconds, a window appears during execution showing the remaining time to wait.

beep

Parameters: time (ms), frequency (Hz)

Description: Causes the computer to issue a tone at the specified **frequency** for the specified amount of **time**.

1.4.3 Do, While, and If Conditions for Sequences

Do, While, and If conditions are imposed on sequences or subsequences by entering the following conditional statements in the Input Parameters column of the sequence name row:

Condition Type	Statement Format	Example Input Parameters
Do	number of times to repeat	20
While	while:logical statement	while:n<=20
If	if:logical statement	if:conc>=50

The prescribed condition is evaluated before each iteration of a sequence.

1.4.4 Running a Sequence

1) Drag the required Sequence/Step name to the **Sequence/Step name** field.
2) Click on the play button . The Sequence will begin to execute and a red bar in the **Work Sequence** panel will indicate the Step that is currently being executed. The play button will transform into a pause button and the stop button will become active. A tab-delimited text file, sequence log.sqlg in the Project folder, is generated sequentially with the Step details and time of execution. This file can be opened in Microsoft Excel. Make sure this file is not open when trying to run a Sequence.

1.4.5 Pausing a Sequence

1) Press the pause button
2) Execution will pause after completion of the current Step and a blue bar will indicate the next Step to execute.
3) You can move the blue bar to a different Step. When you press the play button again, you will be able to select whether to resume at the paused point (the Step after the red highlighted Step) or at the Step which is highlighted with the blue bar. If you choose to use the blue bar to jump to some point in the Sequence that is inside a repeated Sequence (indicated in the Sequence parameter field), then the iterations for the repeated Sequence are advanced to the last iteration.
4) While in Paused mode you can edit parameter fields but should do so with careful consideration of any unintended consequences.

1.4.6 Stopping a Sequence

1) Press the stop button
2) If there is no **stop sequence** (case sensitive, see below), the Sequence will simply be aborted and data acquisition from detectors is stopped. Pumps continue to operate based upon the last command issued to the device.

1.4.7 The "stop sequence"

An orderly stop process for an instrument can be handled by creating a sequence with the name "stop sequence". This sequence would typically include steps to turn off any pumps and any devices that may still be running. When a sequence is aborted, the stop sequence is executed immediately.

1.4.8 Aborted Sequence by Device Fault

A sequence will automatically be aborted if a device receives a faulty command or if the computer loses communication with a device.

1.4.9 Sequence Log File

Each time a sequence is executed, a log of when each individual step is executed is stored in the "sequence log.sqlg" file within the "misc" project folder.

1.4.10 Copying Sequences

To copy a sequence, drag it over to the resource sequence panel, change the name in the field above, and press Rec.

1.4.11 Deleting Steps and Subsequences from Sequences

Steps and subsequences can be deleted from a Sequence in the Work Sequence panel by selecting the occurrence within a Sequence and pressing Ctrl-Delete on the keyboard.

1.4.12 Deleting Steps and Sequences from the Resource Panels

Steps and full Sequences can be deleted from the Resource Steps and Resource Sequences panels respectively, by selecting the occurrence and pressing Ctrl-Delete on the keyboard.

1.4.13 Find and Replace

The **Find and Replace** window is launched when a Step or Sequence that occurs in other Sequences is saved from the Work Sequence panel. The Sequences which contain the modified Step or Sequence are shown in the table of this window. This feature allows you to update all or selected occurrences of a specific Sequence or Step.

1) Indicate in the left hand column which Sequences you would like to perform the find and replace in by clicking in the left column. Selected Sequences will be indicated with a ✓ mark.
2) Click the **Start Targeted Replace** button.
3) The program will proceed through the checked Sequences and locate occurrences of the modified Step or Sequence. These occurrences will be highlighted in Resource Sequences panel.
4) Choose to either **Replace This Occurrence** or **Skip This Occurrence** and the program proceeds to the next occurrence with or without making a replacement.
5) Any Sequences that are changed as a result of the find and replace are automatically saved to disk.

1.4.14 Clearing the Work Sequence Panel

The Work Sequence Panel can be cleared by clicking the **CLR** button

1.4.15 Clearing-Copying to the Work Sequence Panel

The Work Sequence panel can be cleared and a Sequence or Step copied to it by dragging a Sequence or Step into the Sequence/Step Name field.

1.4.16 Opening and Closing Sequences

Clicking on the ⊞ or ⊟ expands or collapses the following branch of the tree. Pressing **Ctrl -** on the keyboard with the mouse arrow over the Resource panels or Work Sequences collapses all the branches in the selected panel. Pressing **Ctrl** + on the keyboard with the mouse arrow over the Work Sequence panel expands all the Sequence branches.

1.4.17 Printing a Sequence

1) Drag the Sequence or Step name to the **Sequence/Step Name** field.
2) Expand the Sequence to show the branches you want printed.
3) To expand the entire Sequence press **Ctrl +**. To collapse the entire Sequence press **Ctrl -**. To expand or collapse a single branch, select it and press + or – key or click on the appropriate ⊞ or ⊟ icon.
4) Right-click within the work sequence panel
5) Select Print

1.4.18 Managing Sequences in Windows

The resource sequences are stored within the corresponding project folder as individual xml files. With FloZF closed, the sequences can be managed from within Windows by removing sequences from the folder or renaming the sequence files to prescribe new names to the sequences. These changes are reflected when the project is next loaded into FloZF.

1.4.19 Exporting Sequences to Excel

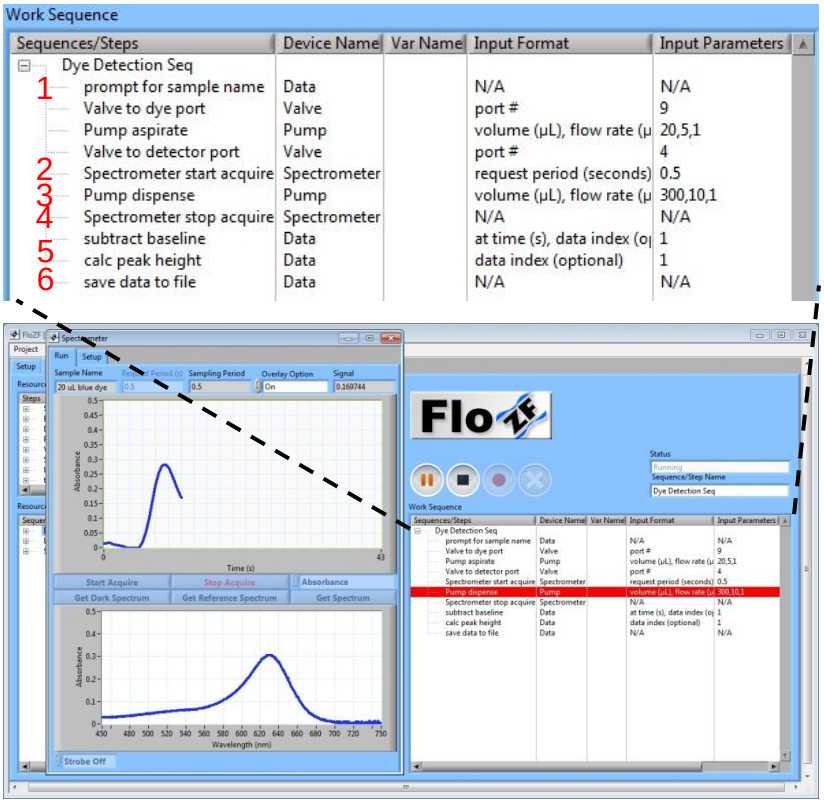
To allow FloZF the required access in Excel, open Excel and in Excel 2003, select Tools, Macro, Security, Trusted Publishers, and check Trust access to Visual Basic Project. In Excel 2010, select File, Options, Trust Center, Trust Center Settings, Macro Settings, and check Trust access to the VBA project object model. Drag the desired sequence to the work sequence panel, right-click within the panel, and select Export to Excel.

2 Data Acquisition

2.1 Tutorial

2.1.1 Building a Data Acquisition Sequence

A typical data acquisition sequence contains the following steps


Work Sequence
Sequences/Steps
Device Name
Var Name
Input Format
Input Parameters
Dye Detection Seq
1 prompt for sample name Data N/A N/A
Valve to dye port Valve port # 9
Pump aspirate Pump volume (µL), flow rate (µ 20,5,1
Valve to detector port Valve port # 4
2 Spectrometer start acquire Spectrometer request period (seconds) 0.5
3 Pump dispense Pump volume (µL), flow rate (µ 300,10,1
4 Spectrometer stop acquire Spectrometer N/A N/A
5 subtract baseline Data at time (s), data index (o 1
6 calc peak height Data data index (optional) 1
save data to file Data N/A N/A
FloZF

1. "prompt for sample name" from the **Data** group of resource steps. This will create a pop-up window for the user to enter the sample name.
2. "start acquire" from the detectors group of resource steps
3. Some step such as the **System** "wait" step that occupies the time during which the detector is acquiring data. In this example the step is the pump "dispense" step.
4. "stop acquire" from the detectors group of resource steps.
5. Data manipulation steps such as "subtract baseline" and "calc peak height" from the Data group.
6. The "save data to file" step, also from the Data group.

2.1.2 Tracking Results

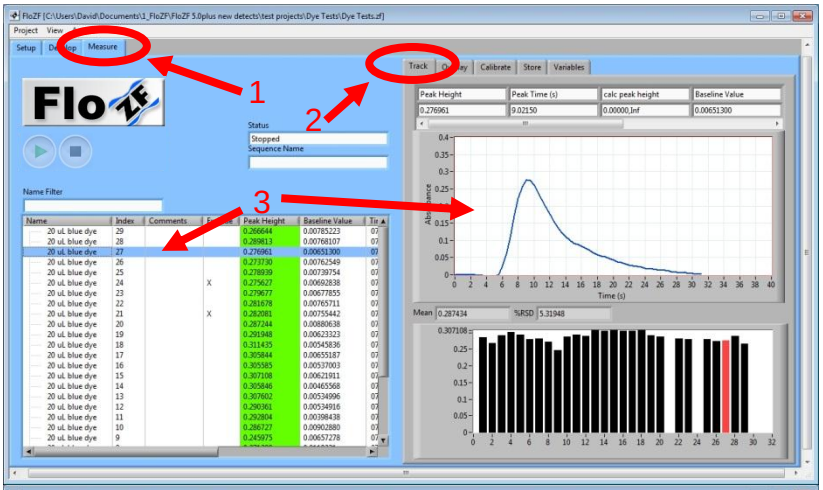
1) Select the **Measure** page of the main window

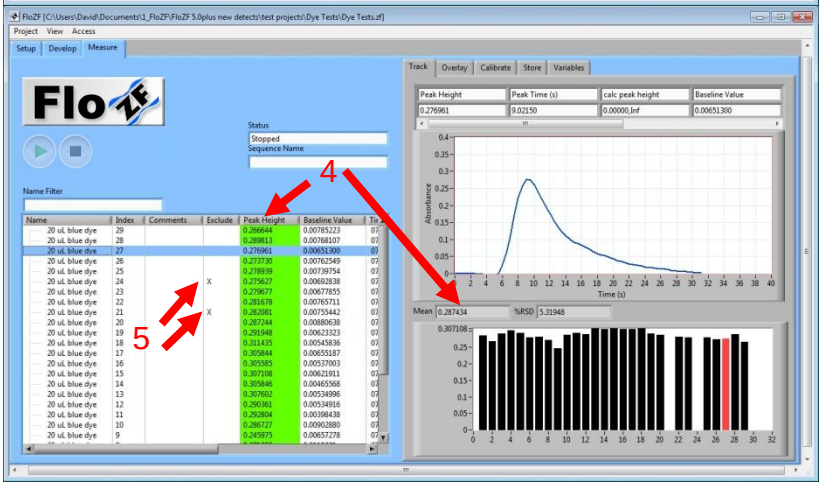
2) Select the **Track** subpage

3) Select the **profile** interest in the data table. The profile appears on the Track subpage with calculated results above.

4) Select the column header of results in the data table to be charted. The corresponding histogram appears on the Track subpage along with the mean and %rsd.

5) Results can be excluded from the histogram and statistics by selecting the cell on the corresponding row in the **Exclude** column once to highlight the row and then a second time to produce an X in the cell.



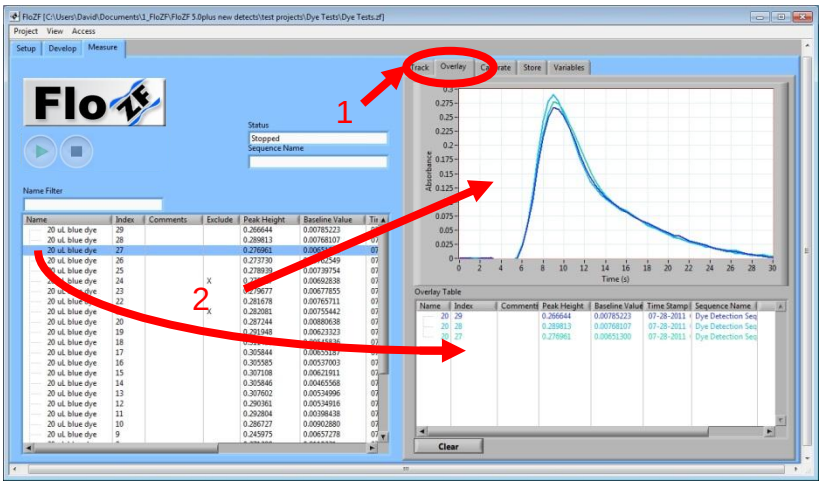


2.1.3 Overlaying Profiles

1) Select the **Overlay** subpage.

2) Drag profiles to be compared from the data table into the **Overlay Table**. The profiles appear

3) To remove profiles from the overlay, select the profile name in Overlay Table and press Ctrl-Delete.



2.1.4 Creating a Calibration Table

1) Select the **Calibrate** subpage

2) Select **Make New**. A window for initializing the table appears

3) Enter a **Calibration Table Name**

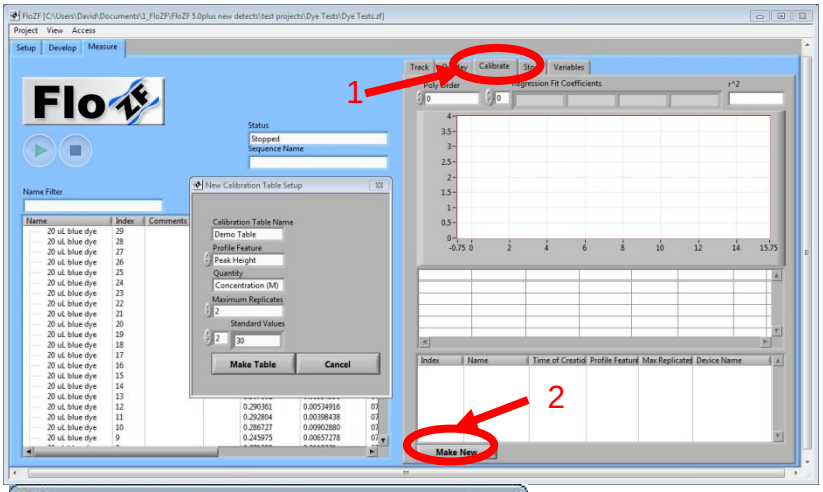
4) Select the **Profile Feature** such as Peak Height that will be used in determining the concentration of samples.

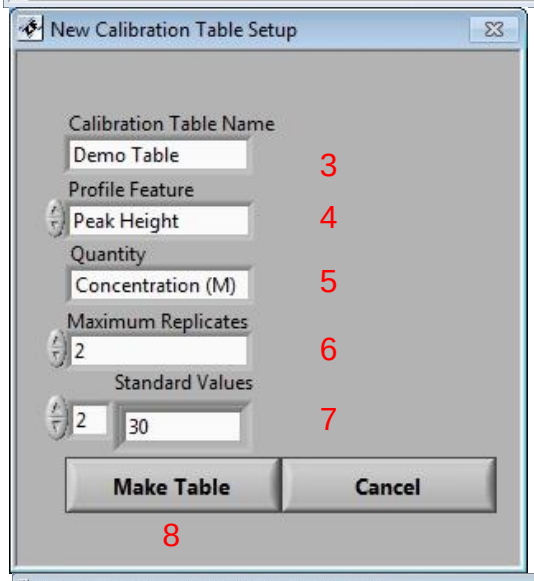
5) Enter the **Quantity** or name for what is to be calculated. Typically this is concentration with appropriately chosen units.

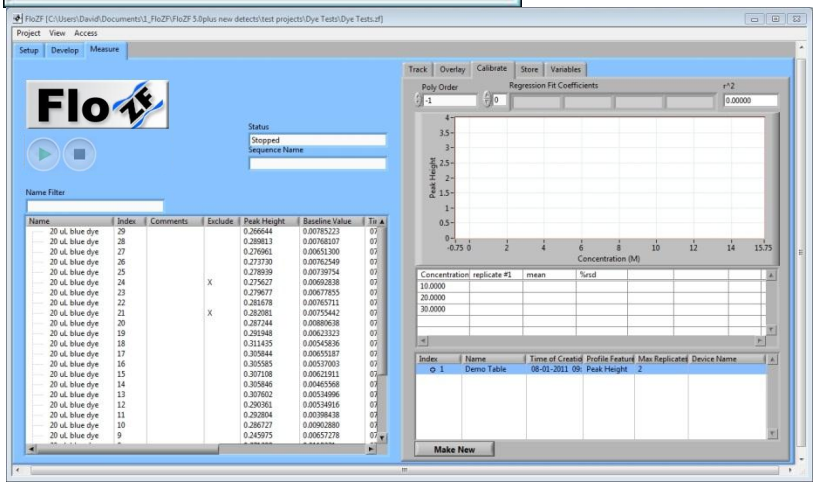
6) Enter the **Maximum Replicates** used in table. The replicates first entered are the first removed after the maximum is reached.

7) Enter the values of the standards that are to be used in the table. Use the index increment on the left to move to different entries. The first entry has an index of zero.

8) Select **Make Table**. A blank table is created in the main window.





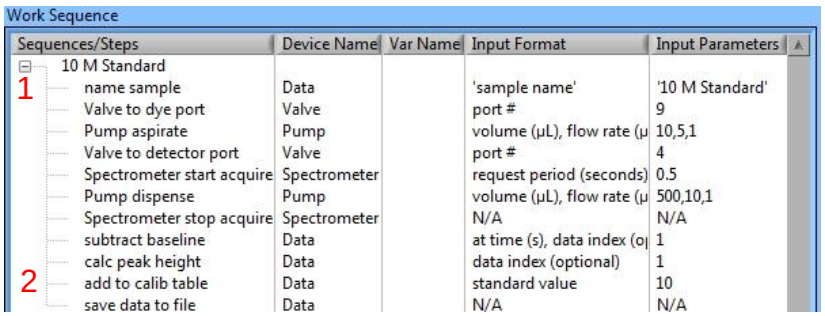


2.1.5 Populating Calibration Tables from Sequences

The following sequences are derived from the sequence example above.

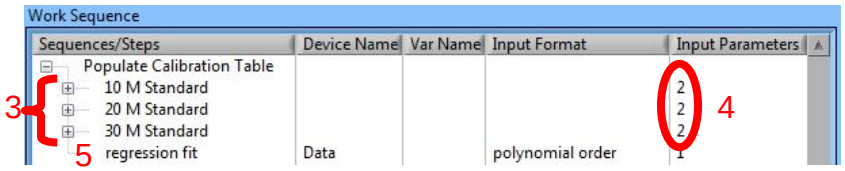
Sequences for each standard typically include

1) The “name sample” step from the data category with the corresponding standard name
2) The “add to calib table” step from the data category with corresponding standard value after the step used to determine the appropriate peak feature



The main calibration sequence includes

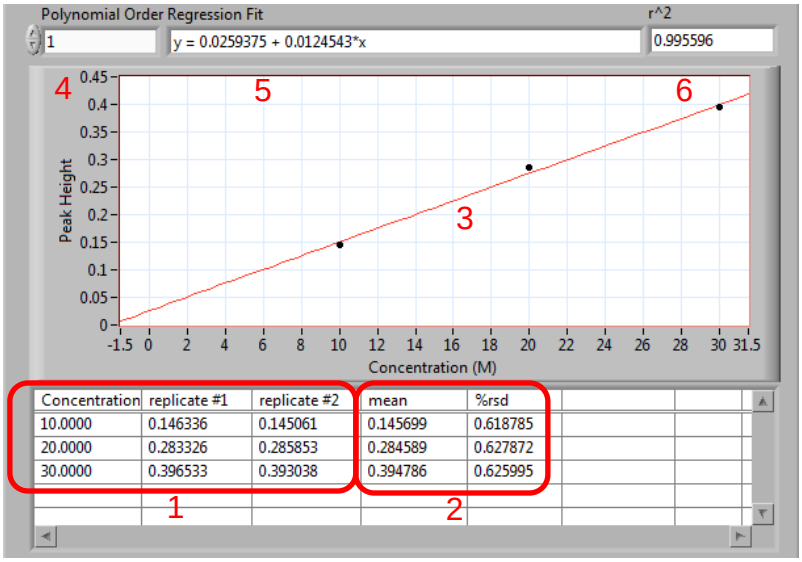
3) Sequences for each standard
4) Repetition of the standard sequences
5) The “regression fit” step to fit a curve to the table data.



2.1.6 Viewing Calibration Tables

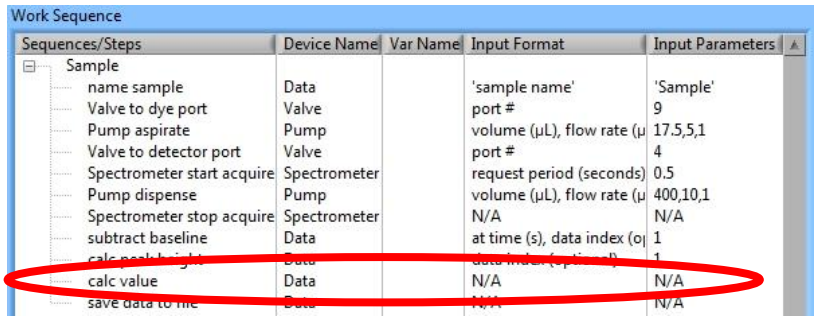
Results of the calibration fit are found on the calibration subpage of the Measure page and include:

1) Individual results from each calibration point
2) Statistics for each standard
3) A plot with the standard means and curve fit
4) Order of the polynomial fit
5) The curve fit
6) The r-squared difference between the curve fit and standard means



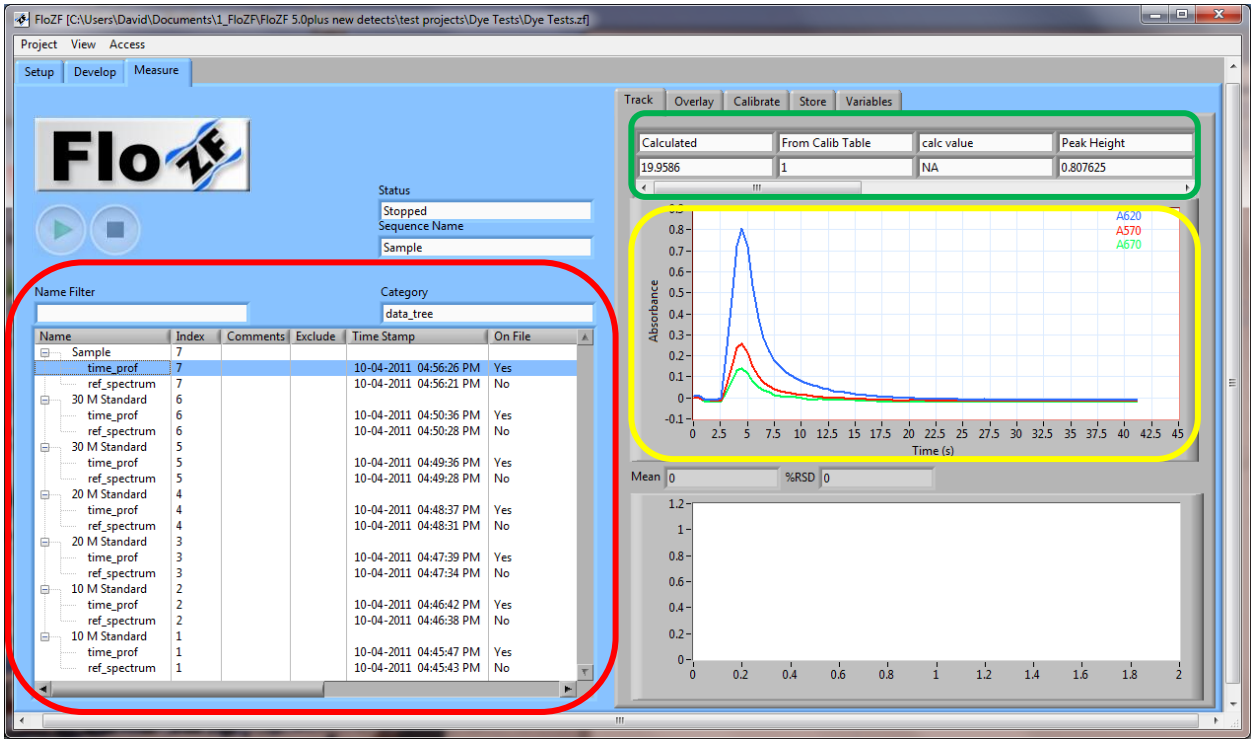
2.1.7 Using Calibration Curves in Sequences

To determine the concentration for a sample use the “calc value” step in a data acquisition sequence after the calculation for the peak feature. The resulting calculated value appears in the data table.



2.2 Profiles

2.2.1 Data Organization



Data acquired from a detector and features associated with the data are stored in three different groups and can be viewed from the **Measure** page of the software.

Profile Plots (Yellow)

Data acquired from a detector, such as time profiles and spectra, are stored as profile plots. Each profile plot has an xlabel, groups of X and Y data, and legends and quantities that correspond to the Y data.

Table Data (Red)

Measurement results are summarized in the table on the left-hand side of the measure page with a row allocated in the table for each profile plot that is obtained from a detector. By default each profile plot is considered a separate sample with its own **sample name** and **sample index** that appear in the first and second column respectively. However, the software makes it possible to associate several related profiles with a single sample by choosing appropriate categories for each category. In the image shown above a spectrometer reference spectrum is stored for sample in the category sample.ref_spectrum, while the time profiles are stored in the category time sample.time_profile. In this case, the sample name appears as a parent to the two category names and the index for all the categories is the same for all the categories within a given sample. Additional data stored in the summary table are any comments the user adds, whether they choose to exclude certain profiles from the statistics calculations, the time stamp for when the data was acquired, indication as to whether the profile has been saved to file or not, and any results that have been calculated using the profiles such as a profile peak height.

Meta Data (Green)

Meta data consists of any data processing commands that were performed on the associated profiles plus the results of those commands.

2.2.2 Data Steps (Excluding Calibration Steps)

Data commands/steps are available once the first detector is added to a project. Some of the commands are intended to operate on the data from a profile plot. Rather than use the name for the plot each time, an **active plot** is defined and its usage is introduced through the commands given below. The ordering of these commands corresponds to the order in which the commands would most likely be used in a sequence.

increment index

Parameters: N/A

Description: This command can be used to indicate a new sample when using multiple plot categories.

name sample

Parameters: ‘sample name’

Description: Specifies the sample name to be used in labeling any data that is acquired after the command has been issued. This name should be listed inside single quotes and should not include any symbols that cannot appear in a file name, such as the symbols \ and /.

prompt for sample name

Parameters: N/A

Description: Launches a window prompting the user to enter a name for the sample. This name is used to label any data that is acquired after the command has been issued. This name should not include any symbols that cannot appear in a file name, such as the symbols \ and /.

start acquire (located with each detector's commands)

Parameters: request period

Description: Starts data acquisition with a sampling period of **request period.**

stop acquire (located with each detector's commands)

Parameters: N/A

Var Name: category name (optional)

Description: Ends the data acquisition and associates the resulting profile plot with the specified category. If the current sample (based on sample index) already has a category by this name or if the sample name has changed, then the sample index is automatically updated and a new sample listing is created. If no category is specified, the sample index is automatically updated, a new sample listing created, and the profile plot is associated with the default "sample" category. This profile plot is now the **active plot** upon which subsequent plot commands will operate. The last five profile plots of any category are automatically stored in memory. The **save data to file** command must be issued to save a profile plot to disk.

make plot active

Parameters: 'plot name'

Description: Makes the entered plot or category the **active plot** on which data operations can be performed. This command is not necessary if the desired plot is already active as a result of the **stop acquire** command.

boxcar average

Parameters: boxcar width

Description: Performs boxcar averaging of the specified width over the active plot data.

set data window

Parameters: min time (s), max time (s)

Definition: Defines a window starting at min time and ending at max time in which the peak height, peak average, or peak area is to be calculated. This window remains set until the command is reissued and defaults to 0 to Inf (infinity) when the program is exited.

subtract baseline

Parameters: at time (s), data index (optional)

Definition: Finds the median value of 7 consecutive data points from the **active plot** and selected data or profile index (default of 1). The 7 data points are chosen so that the middle point is closest in time to the value specified by the parameter **at time**. This median value is subtracted from each initial data point and the modified data are saved in memory. The middle point is saved to the meta data.

calc peak height

Parameters: data index (optional)

Var Name: peak height variable (optional)

Description: Finds the maximum peak height of the active plot and selected data or plot index (default of 1) within the data window (default 0 to infinity) and saves the result to the **peak height variable**, the table data, and the meta data.

calc peak average

Parameters: data index (optional)

Var Name: peak average variable (optional)

Description: Calculates the peak average of the active plot and selected data or plot index (default of 1) within the data window (default 0 to infinity) and saves the result to the **peak average variable**, the table data, and the meta data.

calc peak area

Parameters: data index (optional)

Var Name: peak area variable (optional)

Description: Calculates the peak average of the active plot and selected data or plot index (default of 1) within the data window (default 0 to infinity) and saves the result to the **peak area variable**, the table data, and the meta data.

save data to file

Parameters: N/A

Var Name: file name variable (optional)

Description: Saves the most recently acquired data, modified by any data processing Steps, to a tab-delimited text file in the **active data folder** and to a non-editable xml file. The name of the text file is of the form samplename_index.pfl. Therefore, the samplename should not include any symbols that cannot appear in a filename, such as the \ and / symbols. The resulting filename is saved to the file name variable.

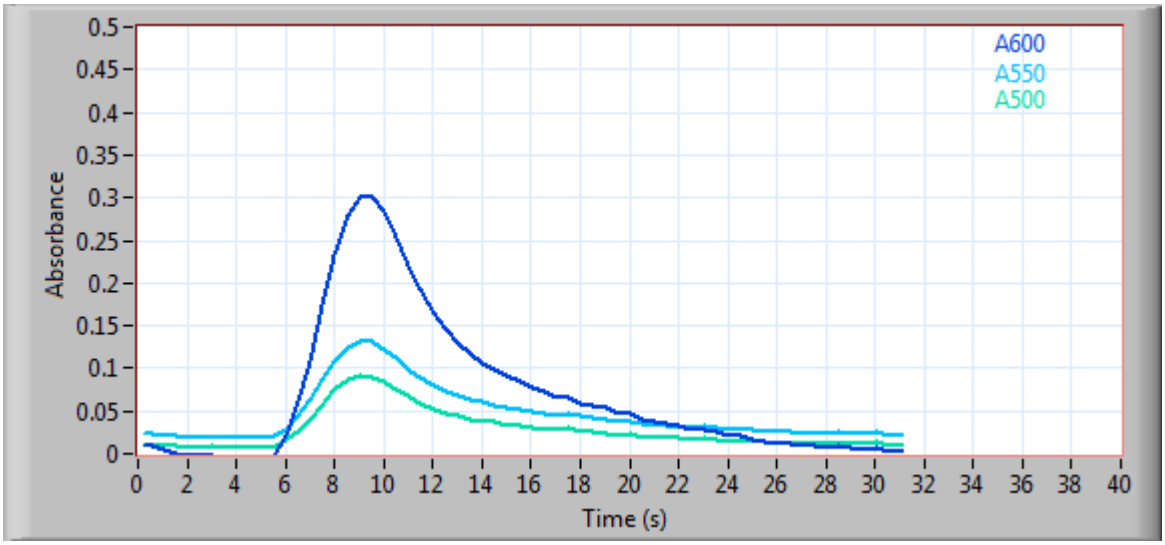
2.2.3 Adjusting Plot Ranges

For most plots, the x and y ranges can be modified by clicking on and editing the minimum and maximum values on the axes.

2.2.4 Plot Variables

The components of a plot can be accessed and modified as variables from either the Variables page or from a sequence. The variable name of each component for a given plot is prefixed by the plots category and index followed by a period. If the index is left out of the variable name, then the current value for the plot index is used. For example, if current plot index is 10, then sample.spectrum.xlabel can be used rather than sample_10.spectrum.xlabel, where sample.spectrum is the category. This convention allows reference to the current plot in a sequence without need for the current index. **CHANGES TO A PLOT AFTER THE PLOT HAS BEEN SAVED ARE NOT AUTOMATICALLY SAVED.**

Each of the plot components are described below and are presented along with results from the following plot. This plot is in the sample category and has the index 10.



xlabel

Description: The **xlabel** is displayed at the bottom of a plot and is stored as a string (text) variable.

Example:

```
>>sample_10.xlabel
Time (s)
```

quantities

Description: The quantities are the units associated with each profile within a plot. The quantities are stored in a single string (text) vector variable. If each profile in a plot has the same quantity, then this quantity is displayed along the y-axis of the plot (ylabel). Otherwise, no label is displayed on the y-axis and the units should be included in the legend.

Example:

```
>>sample_10.quantities
Absorbance Absorbance Absorbance
```

legend

Description: A legend is displayed in the top right corner of a plot when there are two or more profiles on the plot. The legend is stored as string (text) vector variable.

Example:

```
>>sample_10.legend
A600 A550 A500
```

Data:

colors

Description: The RGB colors for each profile on the plot. These colors are stored as a number array with three entries in each row for the red, green, and blue (range 0-255) elements of the corresponding profile color.

Example:

```
>>sample_10.colors
0.000000 65.000000 220.000000
2.000000 191.000000 247.000000
1.000000 220.000000 171.000000
```

X axis data

Description: The x-data against which y-data is plotted. This data is stored as a number vector. The data for a plot are grouped based upon whether they share the same x-axis data or not. There can be multiple groups of x and y data in a plot. Therefore, the corresponding groups for the x and y data are indicated with an index following an underscore.

Example:

```
>>sample_10.X_1
0.250500 0.536500 1.020000 1.522000 2.020000 2.522000 3.020000 3.521500 4.019500 4.521500 5.019500
5.521500 6.019500 6.521500 7.019500 7.521500 8.019500 8.521500 9.023500 9.521500 10.023500 10.521000
11.023000 11.521000 12.023000 12.521000 13.023000 13.521000 14.023500 14.521500 15.019500 15.521500
16.023500 16.521000 17.023000 17.521000 18.023000 18.521000 19.023000 19.521000 20.023000 20.521000
21.023000 21.521000 22.022500 22.521000 23.022500 23.520500 24.022500 24.520500 25.022500 25.520500
26.022500 26.520500 27.022000 27.520500 28.022500 28.520500 29.022500 29.520000 30.022000 30.520000
31.022000
```

Y axis data

Description: The measurement data obtain from a dector. This data is stored as a number array with each row corresponding to a different profile. The index in the variable name corresponds to the index in the associated x-data.

Example:

```
>>sample_10.Y_1
0.009289 0.008299 0.004590 0.000000 -0.001571 -0.002979 -0.003120 -0.003591 -0.006565 -0.006178
-0.006073 -0.001792 0.021558 0.060545 0.109095 0.174264 0.232786 0.279894 0.301374 0.301510 0.282928
0.252987 0.220064 0.191894 0.164026 0.145130 0.128281 0.117185 0.106742 0.097956 0.090544 0.084550
0.077859 0.072188 0.066744 0.066122 0.059726 0.056267 0.053015 0.047096 0.047448 0.039959 0.036792
0.035007 0.031853 0.029187 0.028353 0.025238 0.021683 0.020309 0.016500 0.014106 0.012589 0.011208
0.010365 0.009894 0.007512 0.008915 0.007683 0.004881 0.007128 0.002482 0.002910
0.023512 0.022888 0.020512 0.020874 0.018260 0.019498 0.019658 0.018303 0.017611 0.017920 0.018247
0.020340 0.027588 0.043601 0.062574 0.086996 0.108535 0.124847 0.131728 0.130318 0.122379 0.111666
0.099761 0.088908 0.080210 0.071964 0.067777 0.062443 0.059875 0.056561 0.054191 0.050545 0.048724
0.046587 0.044877 0.046446 0.044917 0.041111 0.040160 0.037311 0.038038 0.034752 0.033908 0.033349
0.031435 0.030893 0.032293 0.031623 0.028922 0.027078 0.027162 0.026013 0.026120 0.025391 0.024193
0.024597 0.023054 0.024690 0.023849 0.022374 0.024833 0.021175 0.022171
0.011360 0.009808 0.009502 0.008455 0.007528 0.007449 0.008559 0.007485 0.007061 0.006564 0.006474
0.006894 0.016374 0.024936 0.039725 0.057312 0.075402 0.087253 0.092163 0.088705 0.083929 0.075877
0.067168 0.059425 0.052518 0.047880 0.044857 0.040819 0.037428 0.036937 0.032737 0.031961 0.030036
0.028003 0.027775 0.030614 0.026480 0.024768 0.022589 0.021350 0.023472 0.019351 0.019043 0.018504
0.018111 0.016835 0.018122 0.016068 0.014017 0.014677 0.013264 0.014145 0.011498 0.010807 0.012036
0.010983 0.012112 0.012791 0.011695 0.010951 0.014423 0.010251 0.009866
```

To obtain a single profile from the Y_1 data, append (data index,:) to the variable name. This example obtains the first profile, which is found in the first row.

```
>>sample_10.Y_1(1,:)
0.009289 0.008299 0.004590 0.000000 -0.001571 -0.002979 -0.003120 -0.003591 -0.006565 -0.006178
-0.006073 -0.001792 0.021558 0.060545 0.109095 0.174264 0.232786 0.279894 0.301374 0.301510 0.282928
0.252987 0.220064 0.191894 0.164026 0.145130 0.128281 0.117185 0.106742 0.097956 0.090544 0.084550
0.077859 0.072188 0.066744 0.066122 0.059726 0.056267 0.053015 0.047096 0.047448 0.039959 0.036792
0.035007 0.031853 0.029187 0.028353 0.025238 0.021683 0.020309 0.016500 0.014106 0.012589 0.011208
0.010365 0.009894 0.007512 0.008915 0.007683 0.004881 0.007128 0.002482 0.002910
```

2.3 Calibration Tables

2.3.1 Data Organization

A calibration table consists the list of standards used, the results for the replicates of each standard such as peak height also referred to as the data set, the statistics on the replicates results, and the regression fit to the results.

2.3.2 Calibration Table Steps

Calibration commands are generated when the first detector is registered and can be found under the Data category in the Resource Steps panel. Rather than use the name for the table data or table regression each time, an **active data table** (result entries in a calibration table) and **active regression table** are defined and need not correspond to the same calibration table. The two tables are indicated in the bottom list of the Calibrate page with a square symbol and rectangle symbol respectively when table are distinct. When the tables coincide this is indicated with a plus symbol. Their usages are introduced through the commands given below. The ordering of these commands corresponds to the order in which the commands would most likely be used in a sequence.

activate table by name

Parameters: 'table name'

Description: This command activates the last active data table and active regression table with the specified **table name**. The active data table and the active regression table need not correspond to the same calibration table since multiple tables can have the same name because they have different identifying indices. This command is typically only necessary if using different table types with different names.

activate table by number

Parameters: table number

Description: Sets the active data table and active regression table to be the calibration table with the index specified in the parameters field.

add to calib table

Parameters: standard value

Description: This command adds the calibration profile feature of the active plot to the active data table. If this table has been used to create a regression and if the regression has been used to calculate values, then a new calibration table is created first, it is designated as the active data table, and the profile feature is added to this table rather than the previously active data table. The designated active regression table is left unchanged in both cases.

regression fit

Parameters: polynomial order

Description: Performs a regression fit of order **polynomial order** on the active data table. This table becomes the active regression table.

calc value

Parameters: N/A

Var Name: calculated value variable name (optional)

Description: Uses the active regression and the most recently calculated corresponding profile feature from the active plot to calculate the desired value. This value is stored in the calculated value variable.

clear calib table

Parameters: N/A

Description: Clears the contents of the active data table.

2.3.3 Accessing Calibration Tables from the Calibrate Page

1) Select the desired calibration table from the list at the bottom of the page.
2) The calibration data appears in the upper table, a plot of the calibration data and regression fit appear in the plot above, and the coefficients for the regression are given at the top of the page.
3) Modifications can then be made to the table as described in the subsections to follow.

2.3.4 Adding Data to Calibration Tables

1) Select the desired calibration table in the bottom table of the page.
2) The data for the table appears in the upper table.
3) Drag the index of the desired profile from the Profile Summary Table into the desired row (based on standard value) and column of this upper table. The entries can not be dropped in columns that would leave a blank entry between results entries.
4) The plot and regression coefficients above the calibration table will update automatically. Any calibration results that are based upon the modified table will also be updated automatically.

2.3.5 Modifying Regressions

Regression fits are performed when the value of **Poly Order** field is changed and whenever modifications are made to the table data.

2.3.6 Clearing Calibration Tables and Deleting Entries

Individual entries can be deleted from a calibration table by selecting the specific entry and pressing **Ctrl-Delete** on the keyboard.

2.3.7 Saving Calibration Tables

Calibration tables saved automatically upon creation and any modification. The tables are saved to tab-delimited text files in the **active data folder** and to non-editable xml files. The name of the text files are of the form calibration_name_index.caf.

2.3.8 Copying a Calibration Table

1) Select the table to copy from the list at the bottom of the Calibration page
2) Right-click within the list
2) Select **Make Copy**

2.3.9 Designating the Active Data and Regression Tables

1) Select the table to activate from the list at the bottom of the Calibration page
2) Right-click within the list
3) Select one of the following options:

   Make Data Active – designates the selected calibration table as the active data table
   Make Regression Active – designates the selected calibration table as the active regression table
   Make Calib Active- designates the selected calibration table as both the active data table and the active regression table

2.3.10 Copying Calibration Tables to New Data Folders

To copy an empty calibration table from one data folder to a new data folder:

1) Select the table to copy from the list at the bottom of the Calibration page
2) Right-click within the list
3) If Copy Blank Table to New Folders is not checked, select it.

2.4 Data Folders

Data folders are created and managed from the Store page within the Measure page. In Windows explorer, individual data folders are located in the path …/project_name/data_folders. The folder in use at a given time is termed the **active data folder** and is indicated with a black diamond in the **Data Folders** list.

2.4.1 Creating a New Data Folder

1) Press the **Create New Data Folder** key in the Data Folders page.
2) In the window that appears, enter the name for the new data folder.
3) The new folder automatically becomes the **active data folder** and the profile and calibration indices are reset to 1.

2.4.2 Designating the Active Data Folder

1) Select the desired folder from the **Data Folders** list.
2) Right-click within the list.
3) Select Activate Selected Folder

4) The data folder profile and calibration summary files are automatically loaded and the profile and calibration indices continue where they left off for the folder.

2.4.3 Deleting a Data Folder

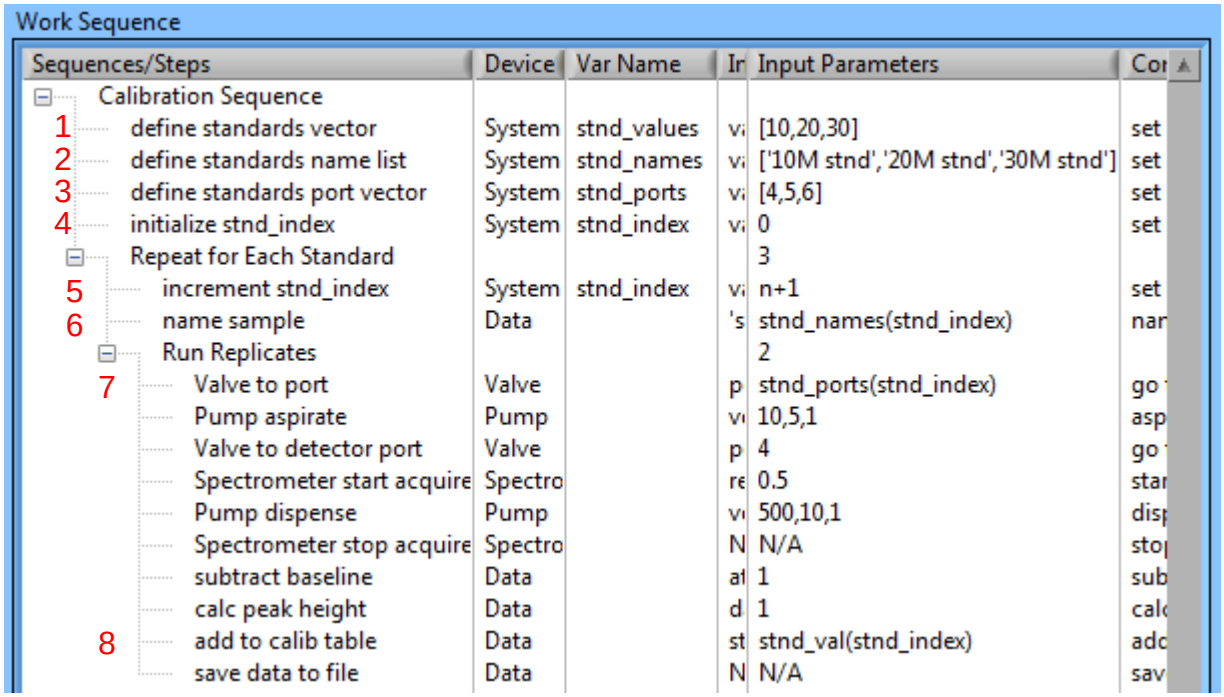
1) Select the desired folder from the **Data Folders** list. It can not be the current active data folder. Press **Ctrl-Delete** on the keyboard.

3 Variables and Excel

3.1 Tutorial

3.1.1 Using Variables within a Sequence

A sequence for populating a calibration table is used here to demonstrate the use of variables within a sequence.


Work Sequence
Sequences/Steps
Device
Var Name
Input Parameters
Calibration Sequence
1 define standards vector System stnd_values [10,20,30]
2 define standards name list System stnd_names ['10M stnd','20M stnd','30M stnd']
3 define standards port vector System stnd_ports [4,5,6]
4 initialize stnd_index System stnd_index 0
Repeat for Each Standard 3
5 increment stnd_index System stnd_index n+1
6 name sample Data stnd_names(stnd_index)
Run Replicates 2
7 Valve to port Valve stnd_ports(stnd_index)
Pump aspirate Pump 10,5,1
Valve to detector port Valve 4
Spectrometer start acquire Spectro 0.5
Pump dispense Pump 500,10,1
Spectrometer stop acquire Spectro N/A
subtract baseline Data 1
calc peak height Data 1
8 add to calib table Data stnd_val(stnd_index)
save data to file Data N/A

A typical calibration sequence using variables is arranged as follows:

At the beginning of the sequence is a section for initializing vector constants and index counters. The specific steps are:

1) The system **set** command to define a **double array** (list of numbers) named **stnd_values** with the numerical values of the stanadard concentrations.

2) The system **set** command to define a **string vector** (list of text fragments) named **stnd_names** with the names for each of the standards. Note that these names are listed inside of single quotes so that they are stored as text. This vector is stored in a variable.

3) The system **set** command to define a **double array** (list of numbers) named **stnd_ports** with the valve port numbers to which the standards are plummed.

4) The system **set** command to initialize a **number** with the name **stnd_index** that acts as an index for which of the three standards is being used. Here **stnd_index** is initialized to the value 0.

Within the main sequence there is a subsequence, listed here as **Repeat for Each Standard**, that is designated to repeat for as many times as there are standards. At the beginning of this subsequence are two steps:

5) The system **set** command to add one to the current value of the counter **stnd_index** and save this value to **stnd_index**. For the first time through the **Repeat for Each Standard** subsequence, this index then has the value one.

6) The **name sample** comand from the **data** category to set the current sample name using the text found in the **stnd_index** entry of the **stnd_names** vector. For the first time through the **Repeat for Each Standard** subsequence the name would be “10M stnd”.

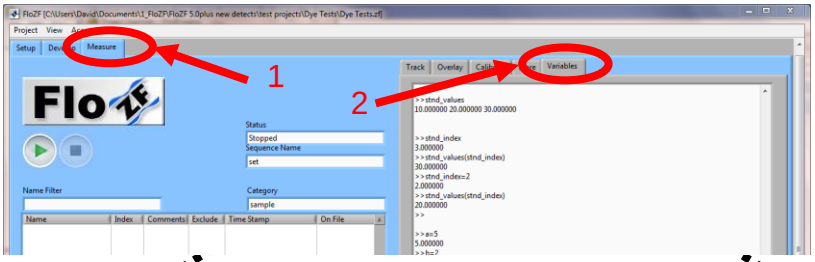
Within the **Repeat for Each Standard** subsequence is the **Run Replicates** subsequence that contains the steps to perform a measurement on each of the standards. This subsequence is designated to repeat for as many replicates of a standard are desired, in this case 2. The steps in this sequence that make use of variables are:

7) The **go to port** command from the Valve category obtains the port number from the **stnd_index** entry of the **stnd_ports** vector. For the first time through the **Repeat for Each Standard** subsequence the port would be 4.

8) The **add to calib table** command from the **data** category obtains which standard value, with which to associate the measurement, from the **stnd_index** entry of the **stnd_ports** vector. For the first time through the **Repeat for Each Standard** subsequence the standard value would be 10.

3.1.2 Viewing Current Variable Values

1) Navigate to the Measure page
2) Select the Variables subpage on the left hand side
3) At the command prompt indicated by the >>, type in the variable name of interest and press Enter. The value for the variable appears.




```
>>stnd_values
10.000000 20.000000 30.000000


>>stnd_index
3.000000
>>stnd_values(stnd_index)
30.000000
>>stnd_index=2
2.000000
>>stnd_values(stnd_index)
20.000000
>>
```

3

3.1.3 Defining New Variables and Evaluating Expressions

On the Variables subpage of the Measure Page you can define variables and perform arithmetic operations.

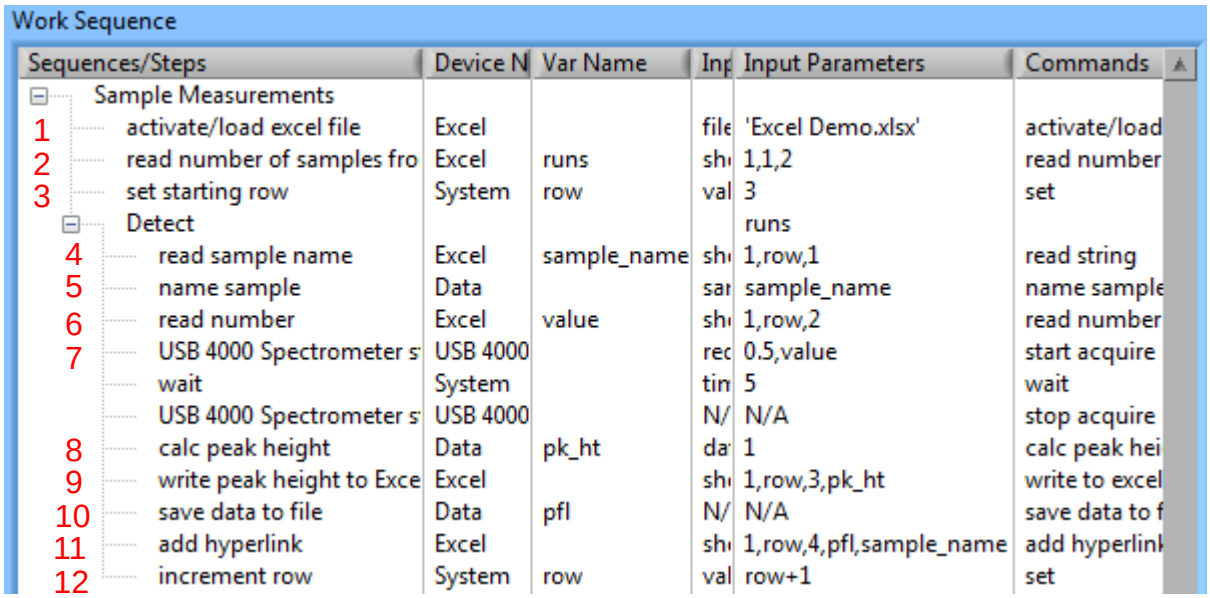
```
>>a=5
5.000000
>>b=2
2.000000
>>c=a/b
2.500000
>>
```

3.1.4 Interfacing with Excel

Shown below is a demonstrative sequence for interfaceing with Excel. This sequence reads a series of sample names and dummy concentration parameters, generates random data about these parameters, calculates concentrations based on the data, and report the data in the original Excel spreadsheet.

	A	B	C	D
1	No of Samples	7		
2	***Name***	***Parameter***	***Peak Height***	***Profile***
3	Sample 1	2		
4	Sample 2	8		
5	Sample 3	1		
6	Sample 4	5		
7	Sample 5	4		
8	Sample 6	9		
9	Sample 7	7		

Excel Spreadsheet Read by Demo Sequence


Work Sequence
Sequences/Steps
Device N
Var Name
Inp
Input Parameters
Commands
Sample Measurements
1 activate/load excel file Excel file 'Excel Demo.xlsx' activate/load
2 read number of samples fro Excel runs sh 1,1,2 read number
3 set starting row System row val 3 set
Detect runs
4 read sample name Excel sample_name sh 1,row,1 read string
5 name sample Data sar sample_name name sample
6 read number Excel value sh 1,row,2 read number
7 USB 4000 Spectrometer s USB 4000 rec 0.5,value start acquire
wait System tin 5 wait
USB 4000 Spectrometer s USB 4000 N/ N/A stop acquire
8 calc peak height Data pk_ht da 1 calc peak hei
9 write peak height to Exce Excel sh 1,row,3,pk_ht write to excel
10 save data to file Data pfl N/ N/A save data to f
11 add hyperlink Excel sh 1,row,4,pfl,sample_name add hyperlink
12 increment row System row val row+1 set

Demo Sequence for Reading Excel Spreadsheet

The demo Excel sequence is arranged as follows.

At the beginning of the sequence is a section for reading constants from the Excel spreadhsheet and for initializing index counters. The specific steps are:

1) The **activate/load excel file** command from the **Excel** category is used to open the Excel file shown above and entitled '**Excel Demo.xlsx**'. This file should be located in the current active data folder if the software is being operated in full mode, otherwise it should be located in the top project folder.

2) The **read number** command from the **Excel** category is used to define a number variable named **runs** that is read from the 1st sheet, 8th row, and 2nd column. This number corresponds to the number of samples to be run which is seven.

3) The system **set** command is used to initialize the **row** counter at 3.

Within the main sequence is the **Detect** subsequence that is designated to repeat for the number of samples. This subsequence is comprised of the following commands:

4) The **read string** command from the **Excel** category to define a string variable named **sample_name** that is read from the 1st sheet, current **row**, and 1st column.

5) The **name sample** command from the **Data** category to associate the data sample name with the **sample_name** read from the Excel file.

6) The **read number** command to define a number variable named **value** that is read from the 1st sheet, current **row**, and 2nd column.

7) The **start acquire** command from the detectors category of steps to generate a random signal about the **value** every half second.

8) The **calc peak height** command from the **Data** category to take the peak value from the randomly generated signal and store it in a number variable with the name **pk_ht**.

9) The **write to excel** command to write **pk_ht** to the 1st sheet, current **row**, and 3rd column of the spreadsheet.

10) The **save data to file** command from the **Data** category to save the profile to file in the active data folder and store the name of this file in a string variable named **pfl**.

11) The **add hyperlink** command from the **Excel** category to create a hyperlink to the file in the active data folder with name stored in the **pfl** variable under the text stored in the **sample_name** variable, in the current **row**, and 4th column of the first sheet.

The Excel spreadsheet after running this sequence is shown below.

	A	B	C	D
1	No of Samples	7		
2	***Name***	***Parameter***	***Peak Height***	***Profile***
3	Sample 1	2	2.919988	Sample 1
4	Sample 2	8	8.874712	Sample 2
5	Sample 3	1	1.95127	Sample 3
6	Sample 4	5	5.94795	Sample 4
7	Sample 5	4	4.93993	Sample 5
8	Sample 6	9	9.998979	Sample 6
9	Sample 7	7	7.965487	Sample 7

3.2 Variables

3.2.1 Assigning Variables

Variables can be defined in FloZF either from the Variables page within the Measure page or from a sequence. On the Variables page the"=" symbol to assign a value to a variable with the desired name. For example, b=7+3 creates a number variable named b with the value 10. In a sequence, the output generated by a step is assigned to the variable listed in the Var Name column. The **set** command with "b" in the Var Name column and 7+3 in the Input Parameters column also creates a number variable named b with the value 10. If a variable with the chosen name already exists, the value and type is changed to match the new designation. Variable names can only be comprised of letters, numbers, periods (.), and underscores (_). Spaces are not allowed in variable names.

3.2.2 Extent of Variables

Once a variable has been created, it remains in memory and can be accessed from anywhere within the software until the software is exited or another project is loaded. Plot data is an exception to this rule since only the five most recent plots of any category automatically remain in memory. If a plot is not saved to file, it cannot be accessed after five more plots in the same category have been obtained.

3.2.3 Variable Types

There are four data types in FloZF:

double (d)

Description: a **number** stored with double precision.

Example:

```
>>a=10
10.000000
```

double array (D)

Description: a two-dimensional **number array** stored with double precision. Arrays are created by entering the desired elements within square brackets [ ] with column entries separated by either a comma or a space and row entries separated by a semicolon ";". Vectors are stored as single row or single column arrays in FloZF.

Examples:

```
>>b=[1,2,3]
1.000000 2.000000 3.000000

>>c=[1 2 3]
1.000000 2.000000 3.000000

>>d=[1;2;3]
1.000000
2.000000
3.000000

>>e=1:7
1.000000 2.000000 3.000000 4.000000 5.000000 6.000000 7.000000

>>f=[1,2,3;4,5,6]
1.000000 2.000000 3.000000
4.000000 5.000000 6.000000

>>
```

string (s)

Definition: a string is a collection of **text**. Strings are created with the text entered within single quotes to distinguish the string from a variable name.

Example:

```
>>sample_name='CN MD'
CN MD
```

string array (S)

Definition: an array of **text** fragments. Arrays are created by entering the desired elements within square brackets [ ] with column entries separated by either a comma or a space and row entries separated by a semicolon ";". Vectors are stored as single row or single column arrays in FloZF.

```
>>months=['Oct','Nov','Dec']
Oct Nov Dec

>>states=['WA';'OR';'CA']
WA
OR
CA

>>colors=['Red','Green','Blue';'Purple','Yellow','Brown']
Red Green Blue
Purple Yellow Brown
```

3.2.4 Arithmetic Operators

The following variables will be used to demonstrate arithmetic operators

variables

```
>>a=2
2.000000
>>b=4
4.000000
>>C=[1,2,3]
1.000000 2.000000 3.000000

>>E=[4,5,6]
4.000000 5.000000 6.000000
```

addition

```
>>f=a+b
6.000000
>>G=a+C
3.000000 4.000000 5.000000

>>H=C+E
5.000000 7.000000 9.000000
```

subtraction

```
>>j=a-b
-2.000000
>>K=C-E
-3.000000 -3.000000 -3.000000

>>L=a-C
1.000000 0.000000 -1.000000
```

multiplication- note that multiplication of two arrays here is element by element and denoted by .*

```
>>m=a*b
8.000000
>>N=C.*E
4.000000 10.000000 18.000000

>>P=a*C
2.000000 4.000000 6.000000
```

division

```
>>q=a/b
0.500000
```

combined

```
>>R=(2+(4*a-6*b))*C
-14.000000 -28.000000 -42.000000
```

3.2.5 Boolean Operators

The Boolean values **True** and **False** are represented in FloZF by 1.0 and 0.0. The following variables are used in the following table to demonstrate arithmetic operators

variables

```
>>a=2
2.000000
>>b=5
5.000000
>>c=3
3.000000
>>e=7
7.000000
```

Boolean Operator	Result is 1.0 (True)	Result is 0.0 (False)
equality	a==2	a==b
inequality	a~=b	a~=2
greater than	b>a	a>b
less than	a<b	b<a
greater than or equal to	b>=5	b>=5.1
less than or equal to	a<=2	a<=1.9
and	a<b&c<e	a<e&b<c
or	a<e|b<c	a<2|b<4

3.2.6 Functions

The following variables will be used to demonstrate arithmetic operators

variables

```
>>a=2
2.000000
>>b=4
4.000000
>>C=[1,2,3]
1.000000 2.000000 3.000000

>>E=[4,5,6]
4.000000 5.000000 6.000000
```

if

Format: result = if(Boolean expression, true result, false result)

Description: Evaluates the Boolean expression and returns the true result if the expression is true, otherwise it returns the false result

Examples:

```
>>s=if(a>b,2*a,6*b)
30.000000
>>R=if(a<=b,C,E)
1.000000 2.000000 3.000000
```

num2str

Format: result_string = num2str(number)

Description: Converts a number to a string (text) presentation of the number

Examples:

```
>>num2str(a)
2
```

strcat

Format: result_string = strcat(string_vector)

Description: Concatenates the strings within the string vector

Examples:

```
>>string1='combine'
combine
>>string2=' these '
 these
>>string3='together'
together
>>strcat([string1,string2,string3])
combine these together
```

rsd

Format: result = rsd(double_vector)

Description: calculates the percent rsd of the double vector

```
>>rsd(C)
50.000000
```

3.2.7 Variables Commands

set

Parameters: value or expression

Var Name: variable name

Description: Evaluates the expression or number in the Input Parameters column and stores the result to a variable addressed by the **variable name**. See the section on Variables and Excel for further detail.

get date in seconds

Parameters: N/A

Var Name: date stamp variable name

Description: Gets the elapsed time in seconds since 12:00 a.m., Friday, Jan 1, 1904 and stores it in the **date stamp variable** as a number.

get date stamp

Parameters: N/A

Var Name: date stamp variable name

Description: Gets the current data and time stamp and stores it in the *date stamp variable* as a string with the format MM-DD-YYYY HH:MM:SS AM/PM

3.3 Excel

3.3.1 Excel Commands

Excel commands are generated when the first device is registered and can be found under the Excel category in the Resource Steps panel. To access Excel files from a sequence, they must be in the **active data folder** for the full version of the software or in the topmost program folder in demo mode. Rather than use the name for the plot each time, an **active excel file** is defined and it's usage is introduced through the commands given below. The ordering of these commands corresponds to the order in which the commands would most likely be used in a sequence.

copy excel file

Parameters: 'original_file.xls',' duplicate_file.xls'

Description: Copies the original_file.xls to the duplicate_file.xls

activate/load excel file

Parameters: 'file.xls'

Description: Opens **file.xls** in Excel and makes this the **active excel file**. If the file is already open, it simply becomes the **active excel file**.

create and activate excel file

Parameters: 'filename.xls'

Description: Creates a blank file and saves it with the name **filename.xls** in the active data folder (or topmost project folder for Demo Mode). This becomes the **active excel file**.

set cells background color

Parameters: sheet#, start row#, start col#, end row#, end col#, color

Description: Sets the background **color** of the cells in the specified **sheet** of the **active excel file** starting at the **start row** and **start column** and ending at the **end row** and **end column**. The options are 'no' for no color, 'w' for white, 'r' for red, 'g' for green, and 'b' for blue.

read number

Parameters: sheet #, row #, col#

Var Name: number variable

Description: Reads a number from the **sheet, row, and column** of the **active excel file** and saves it into the **number variable**.

read number array

Parameters: sheet#, start row#, start col#, end row#, end col#

Var Name: number array variable

Description: Reads a number array from the specified **sheet** of the **active excel file** located between the **start row** and **start column** and the **end row** and **end column** and stores it in the **number array variable**.

read string

Parameters: sheet #, row #, col#

Var Name: string variable

Description: Reads a string (text) from the **sheet, row, and column** of the **active excel file** and saves it into the **string variable**.

read string array

Parameters: sheet#, start row#, start col#, end row#, end col#

Var Name: string array variable

Description: Reads a string array from the specified **sheet** of the **active excel file** located between the **start row** and **start column** and the **end row** and **end column** and stores it in the **string array variable**.

write to excel

Parameters: sheet#, start row#, start col#, any data

Description: Writes any data type to the specified **sheet** of the **active excel file** starting at the **start row** and **start column**.

add hyperlink

Parameters: sheet #, row #, col #, 'filename','text to display'

Description: Creates a hyperlink on the specified **sheet** of the **active excel file** in the cell specified by the **row** and **column.** The hyperlink has the text **text to display** and is linked to the file in the **active data folder** (or topmost project folder for Demo Mode) with the name **filename**.

save excel file

Parameters: N/A

Description: Saves the **active excel file**.

close excel file reference

Parameters: N/A

Description: Closes the **active excel file**.

4 Appendix A: Installation & Updates

4.1 System Requirements

Windows XP, Vista 32 Bit, or Windows 7 32 Bit (**English versions**)

512 MB (Windows XP) or 2 GB (Vista and Windows 7) of memory

Screen resolution 1280×768 or greater (smaller will be inconvenient)

2 x USB or serial ports

5 GB disk space (200 MB for program)

4.2 Installation

Installation of FloZF is initialized by launching the setup.exe found on the installation CD. FloZF is based on LabVIEW by National Instruments (NI), and is dependent on a specific version of the Labview runtime library and Visa runtime library. If the software has been installed from disk, it can be assumed that the correct library has been included on the disk. Subsequent updates may necessitate a different runtime library. It is worth verifying that you have the correct version of these libraries. It is suggested that you choose the default installation locations for both the NI libraries and for the FloZF files in C:\Program Files\. After installing the software, it is necessary to reboot your computer. Other libraries and drivers may also be necessary depending on your specific hardware configuration. In these cases the necessary software is indicated in Appendix B: Device .

For Windows Vista and Windows 7, the permissions for the folder C:\Program Files\FloZF must be changed to allow the software to write to the folder. Right-click on the on folder icon, then select Properties, Security, Edit, then select Users in the window above and check Write in the Allow column in the window below. Select OK to complete the settings change.

4.3 Device Drivers and LabVIEW Libraries Compatibility Matrix

FloZF Release	LabVIEW Runtime Engine	NI VISA Runtime Engine	Ocean Optics Omnidriver+SPAM
4.2	8.5	4.2	Beta
4.3	8.6	4.4.1	Beta
5.0.0.0	2009 SP1	4.6	Beta
5.0.0.422	2009 SP1	4.6	1.5
5.0.0.596	2009 SP1	4.6	1.6

4.4 First Time Launching FloZF

FloZF can be launched from the icon created on the desktop, from the programs list of the start menu, and by selecting the top-level FloZF project files (***see below***). In the first two cases, the last project that was accessed is automatically loaded. A prompt for a Software Key will appear if it is the first time that the software has been launched.

4.5 Obtaining a software key

FloZF can be operated in “Demo Mode” for controlling pumps and valves or in “Full Mode” for controlling pumps, valves, detectors, IO, and other devices. When FloZF is started for the first time a code (Computer ID) is generated specific to your computer. You will be prompted to provide a corresponding Software Key to allow access to “Full Mode”. If you choose to operate in “Demo Mode”, simply select the corresponding button without entering a Software Key. To obtain a Software Key you will need to purchase a license and email the Computer ID to your sales representative. The key is entered at the prompt and tested by pressing the **Test Key** button. This is a one-time operation and links the installed software to the specific computer on which it is installed.

4.6 Switching to Full Mode from Demo Mode

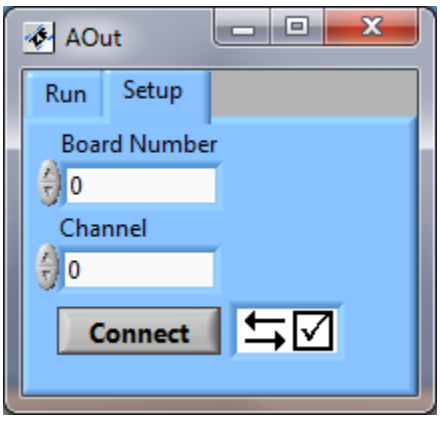
You can switch to full mode at any point by selecting Access, then Software Key in the menus at the top of the main screen and following the instructions above.

5 Appendix B: Device interfaces, setup and commands

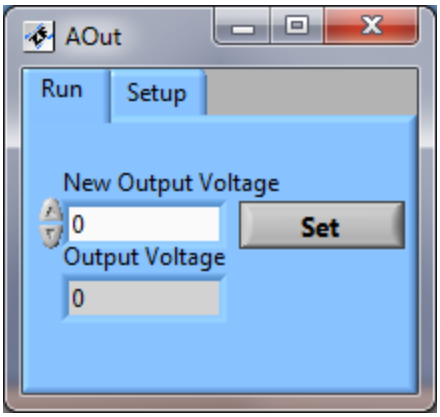
5.1 Analog Output

5.1.1 USB Analog output (Measurement Computing)

Communication with the Measurement Computing DAQ card is performed via USB. Before the device can be operated from within FloZF, the InstaCal must be installed and configured on the computer as described in the USB DAQ (Measurement Computing) subsection.



The communication settings to be entered are the **Board Number** and the **Channel Number** (0 or 1). Communication is tested by pressing **Connect** on the Setup page of the Instrument Panel, and successful communication produces the ⇆☑ symbol.



The analog output can be operated "manually" by entering the desired voltage into the **New Output Voltage** field and pressing **Set**. The voltage at any time is displayed in the **Output Voltage** field.

The only step that is generated in the Resource Steps panel when you register this device is:

set output voltage

Parameters: output voltage

Description: Applies the **output voltage** to the selected channel

5.2 Detectors

5.2.1 General serial detector

5.2.2 Spectrometer (Ocean Optics)

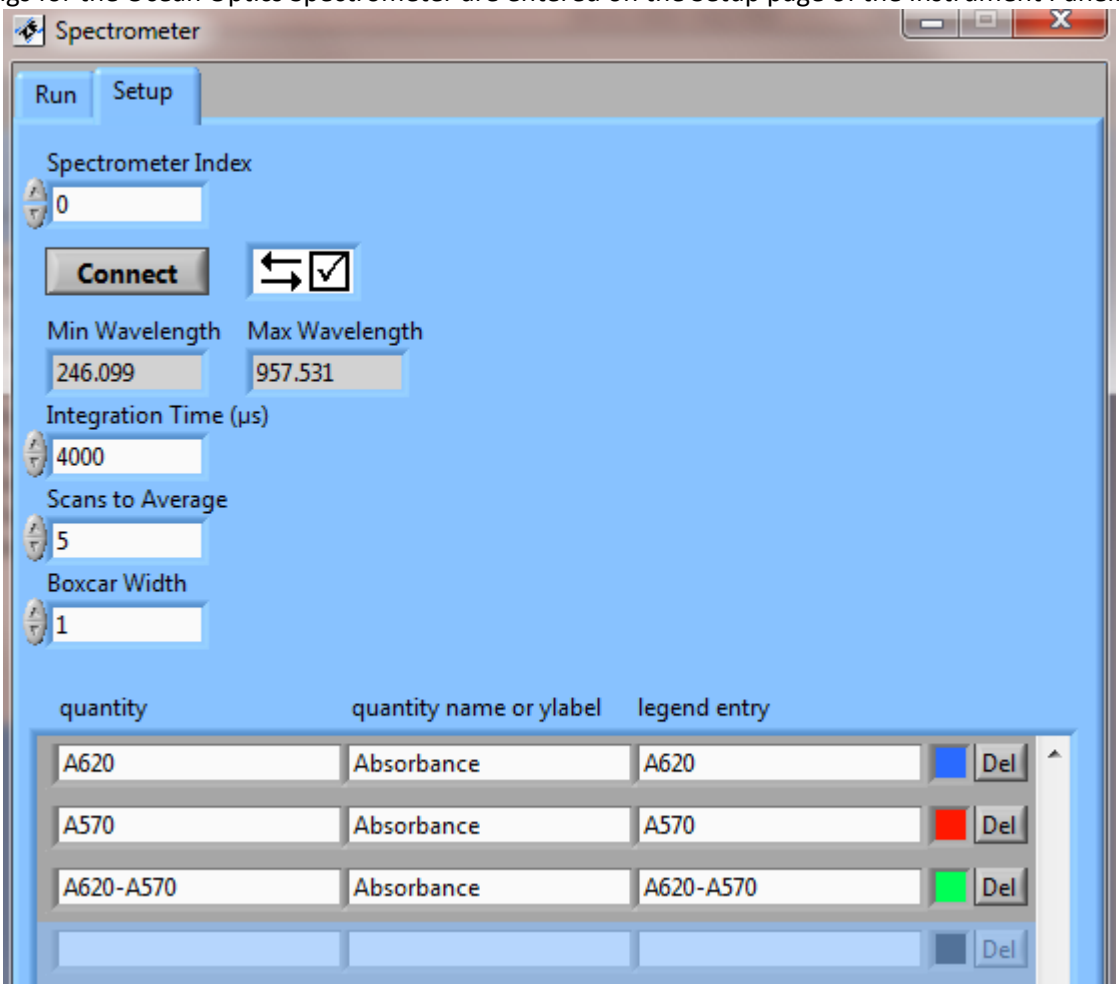
Communication with Ocean Optics spectrometers is performed via USB. Before Ocean Optics spectrometers can be operated from within FloZF, the following setup steps must be taken.

1) Unplug the Ocean Optics spectrometer from the computer if it is connected
2) Open a dos window. In XP this requires selecting **start**, **Run**, entering "command" and pressing **OK**. In Vista this requires typing "command" into the field above **start** and pressing **Enter** on the keyboard.
3) Check which version of Java (if any) is installed by issuing the command "java –version" at the prompt.
4) If an error message appears, indicating Java is not installed, or if the version of Java starts with 1.5_...., then you will need to download and install the latest version of Java from http://www.java.com/en/download/manual.jsp.
5) Refer to the drivers compatibility chart above and Window's Add or Remove Software listing for the Ocean Optics Omnidriver to see whether you have the correct version of the driver installed.
6) If the OO USB setup.exe is listed in Windows then the Beta version of the Omnidriver is installed and you will need to take the following steps:
    a. Uninstall OO USB setup

b. Remove the system variable OOI_HOME. . This requires right-clicking on **My Computer**, selecting **Properties, Advanced system settings** (Windows 7), then the **Advanced** tab, and then the **Environment Variables** button. In the **System variables** panel scroll down to find and select "OOI_HOME". Click Delete and then OK to finish.

7) Run the installer for the version of Ocean Optics OmniDriver+SPAM listed in the compatibility matrix.

Settings for the Ocean Optics Spectrometer are entered on the Setup page of the Instrument Panel.



Spectrometer (Ocean Optics) Instrument Panel: Setup page

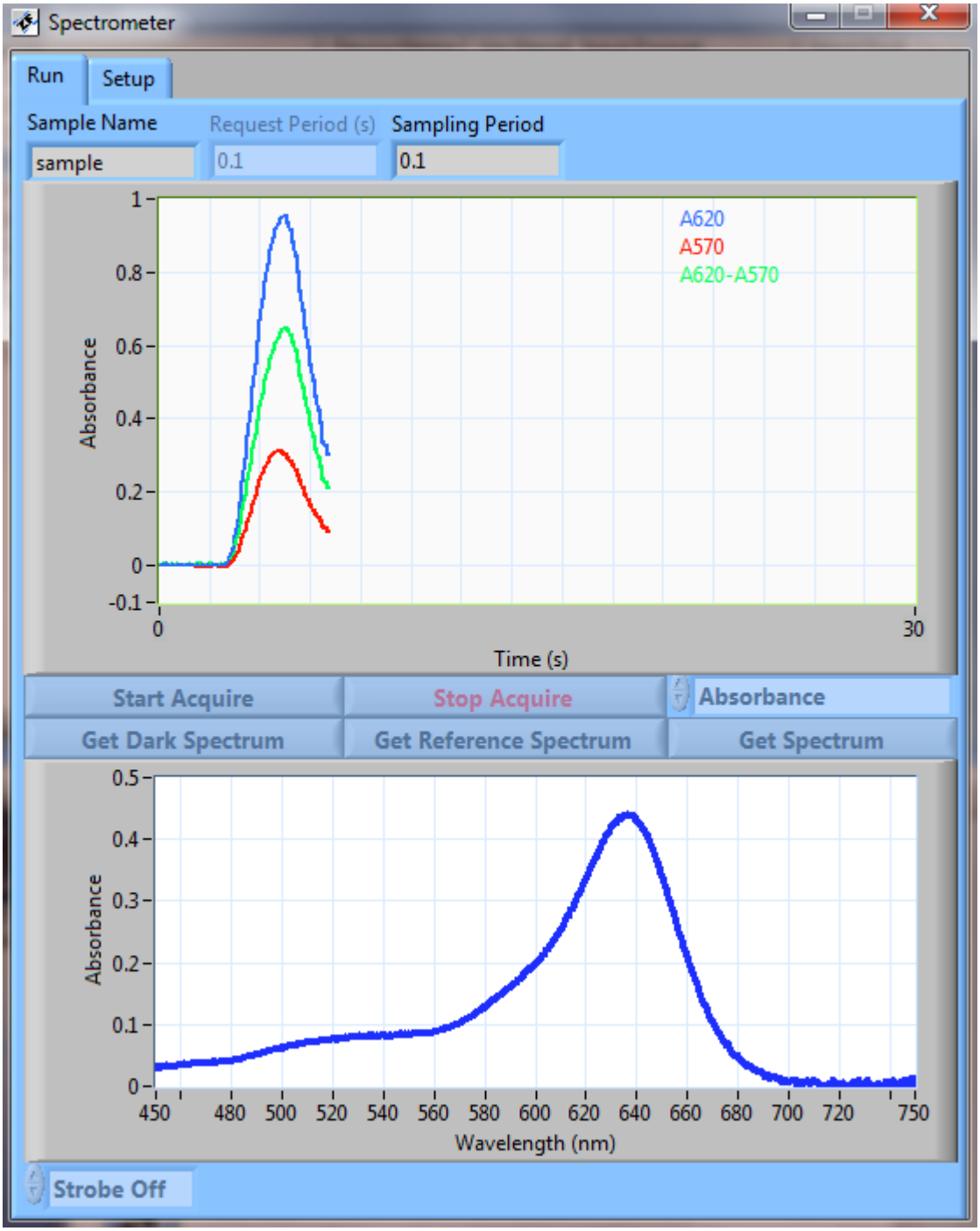
The **Spectrometer Index** is the only communication setting that must be applied. If there is only one spectrometer connected, this index should be 0. Communication is tested by pressing **Connect** on the Setup page of the Instrument Panel, and successful communication produces the ⇆☑ symbol.

The **Min Wavelength** and **Max Wavelength** are read from the spectrometer upon establishing communication. Below these wavelength indicators are settings for how the signal is to be acquired and processed. The first three of the settings are the **Integration Time (µs)**, **Scans to Average**, and **Boxcar Width**. These settings pertain to how a full spectrum is obtained and processed. The spectrometer CCD

acquires light for the specified **Integration Time** and reports the results back to the computer as a single scan. When the number of **Scans to Average** have been received, these scans are averaged and a boxcar average having **Boxcar Width** (specified in number of CCD pixels) is performed over the resulting spectrum.

At the bottom of the setup page is a table for specifying what combination of wavelength results to monitor over time and how to display the results. The first column, **quantity**, is used to indicate what combination of wavelength results to monitor. In this column, the **Absorption**, **Irradiance**, or **Fluorescence** for a specific wavelength is indicated with a prefix of **A**, **I**, or **F**, respectively before the wavelength in nanometers. Only whole numbers are accepted for the wavelengths. Algebraic expressions or combinations containing these wavelength results, such as A620-A570, can be used.

The spectrometer can be operated "manually" from the Run page of the Instrument Panel. **Dark Spectrum** and **Reference Spectrum** are acquired and stored by pressing the respective buttons on the middle of the page. The light source must be switched off to acquire the Dark Spectrum. If a light source is connected directly to the spectrometer, then it can be turned on and off with the control at the bottom left of the panel. **Irradiance**, **Absorbance**, or **Fluorescence** (Irradiance-Reference Spectrum), are acquired by selecting the corresponding choice at the middle-right of the page and then pressing the **Get Spectrum** button below. All acquired spectra are displayed in the plot at the bottom of the page. Time profiles of the quantities chosen on the Setup page are acquired by pressing the **Start Acquire** button. As the profiles are acquired, they are displayed in the top plot and the full spectrum (chosen in the control above the **Get Spectrum** button) is displayed in the bottom plot. If only a single quantity has been chosen on the setup page, then the current value is displayed in the **Signal** field as the signal is acquired. A history of the signals is also displayed for this case if the **Display Overlay** option is chosen. The history can be cleared by right-clicking and selecting "clear history".



Spectrometer (Ocean Optics) Instrument Panel: Setup page

Data acquired directly from the Instrument Panel (rather than in a Sequence) cannot be saved.

The Steps that are generated for the Spectrometer (Ocean Optics) in the Resource Steps panel when you register this device are:

get dark spectrum, get reference spectrum, and get spectrum

Parameters: N/A

Var Name: category name (optional)

Description: Obtains an irradiance spectrum and for the dark and reference spectrum stores it in the computer memory as the detector's dark or reference spectrum, respectively. For the get spectrum command, the measurement that is reported corresponds to the selection above the **Get Spectrum** button. If no **category name** is specified, then these commands perform no additional actions. If a **category name** is specified then the resulting spectrum is saved as a plot in the corresponding category. If the current sample (based on sample index) already has a category by this name or if the sample name has changed, then the sample index is automatically updated and a new sample listing is created. The last five profile plots of any category are automatically stored in memory. The **save data to file** command must be issued to save a profile plot to disk.

set wavelengths

Parameters: wavelength1, wavelength2, wavelength3,……

Description: Populates the formula table on the setup page with the listed wavelengths according to the measurement type chosen on the Run page. For example, if the wavelengths are 470, 520, 600 and Absorbance is chosen on the setup page, then the first column would be A470, A520, and A600 on the first, second, and third row respectively.

get data point

Parameters: N/A

Var Name: data variable name

Description: Obtains a single spectrum from the spectrometer and generates the quantities found in the formula table of the detector setup page. The result is stored either as a number or number array with the **data variable name**.

set formulas

Parameters: formula array (S), colors array (D)

Description: Uses the **formula array** and **colors array** to populate the formula table on the Setup page of the detector. The **formula array** is a string array that contains the text to go into the first three columns of the table. To populate the formula table as shown above, the formula array would be ['A620','Absorbance','A620';'A570','Absorbance','A570','Absorbance','A570';'A620-A570','Absorbance','A620-A570']. The **colors array** is a number array with three entries in each row for the red, green, and blue (range 0-255) elements of the corresponding profile color. To populate the colors in the formula table as shown above, the colors array would be [0 65 220;2 191 247;1 220 171];

start acquire

Parameters: N/A

Description: Starts data acquisition with a sampling period of **request period** (in seconds) or (Integration Time) times (Scans to Average), whichever is longest. The Sequence continues on immediately to the next Step after this Step has been issued. Frequently this will be a wait Step.

stop acquire

Parameters: N/A

Var Name: category name (optional)

Description: Ends the data acquisition and associates the resulting profile plot with the specified category. If the current sample (based on sample index) already has a category by this name or if the sample name has changed, then the sample index is automatically updated and a new sample listing is created. If no category is specified, the sample index is automatically updated, a new sample listing created, and the profile plot is associated with the default "sample" category. This profile plot is now the **active plot** upon which subsequent plot commands will operate. The last five profile plots of any category are automatically stored in memory. The **save data to file** command must be issued to save a profile plot to disk.

turn strobe on/off

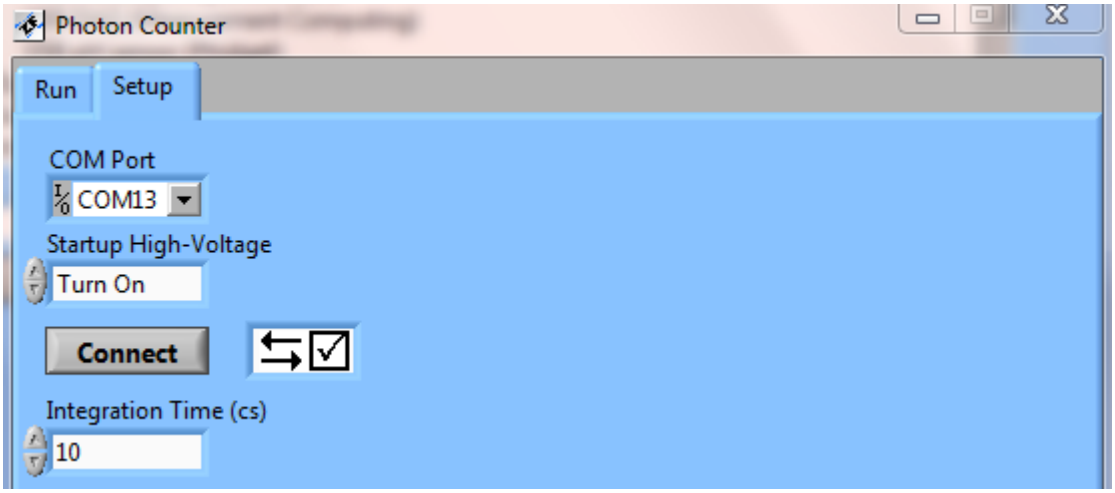
Parameters: N/A

Description: If a light source is connected directly to the spectrometer, then it can be turned on and off with these commands.

5.2.3 Photomultiplier tube (Hamamatsu) & Photon counter (Electron Tubes)

Communication settings for the Phomultiplier (Hamamatsu) are entered on the Setup page of the Instrument Panel. The only required communication setting is the **COM Port**. If the high-voltage is to be turned on upon connecting with the photomultiplier, then **Startup High-Voltage** should be set accordingly.

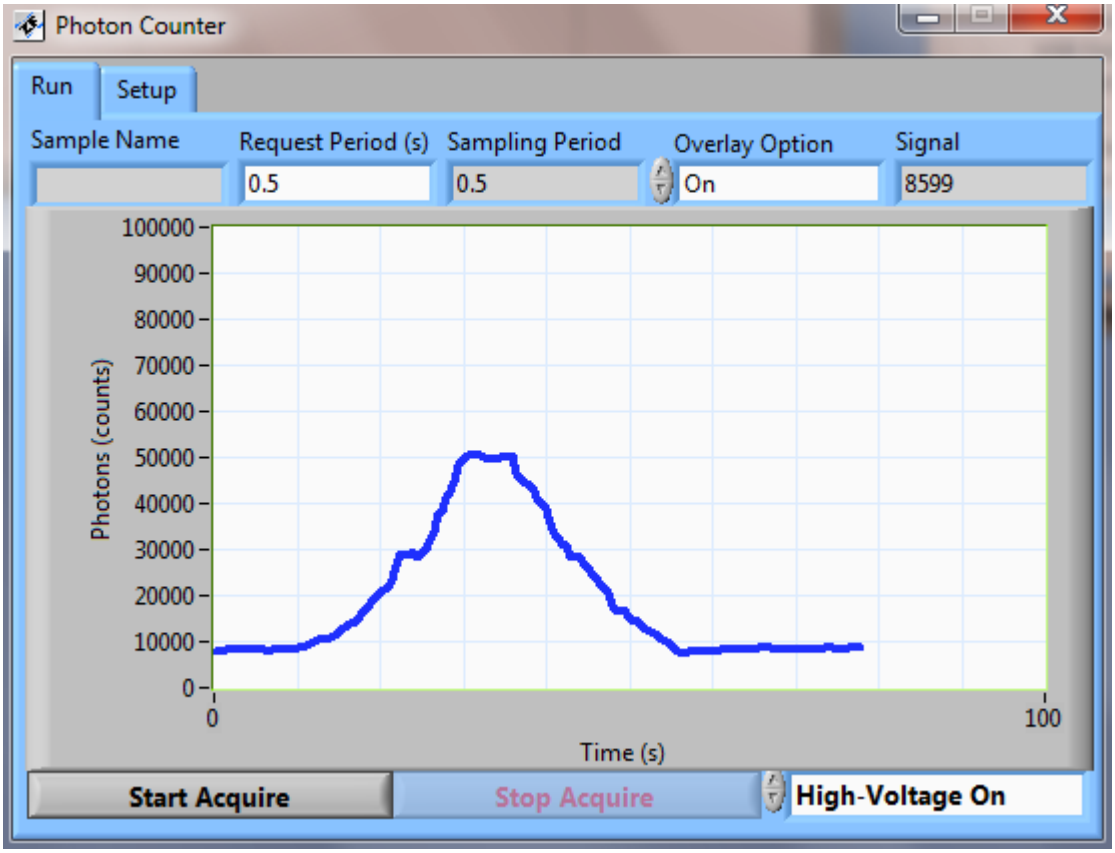
Communication settings for the detector are entered on the Setup page of the Instrument Panel (**Error! Reference source not found.**). The only required communication setting is the **COM Port**. If the high-voltage is to be turned on upon connecting with the photon counter, then **Startup High-Voltage** should be set accordingly.



Photomultiplier tube (Hamamtsu) & Photon counter (Electron Tubes) Instrument Panel: Setup page

When the **Connect** button is pressed, two way communication with the device is tested and the Instrument panel displays the ✕ in the event of unsuccessful testing and ⇆☑ for successful testing.

The **Integration Time (cs)** is the time in centi-seconds over which light is accumulated by the detector for each data point.



Photomultiplier tube (Hamamtsu) & Photon counter (Electron Tubes) Instrument Panel: Run page

Data can be acquired from the Run page of the Instrument Panel if a Sequence is not currently being executed. Data acquisition is initiated and terminated with the **Start Acquire** and **Stop Acquire** buttons, respectively. The number of photon counts at any given time is displayed in the **Signal** field. The high-voltage can be turned on or off with the control in the lower right corner.

Data acquired directly from the Instrument Panel (rather than in a Sequence) cannot be saved. Additionally, only data acquired from a Sequence can be recalled with the overlay options.

The Steps that are generated for the detector in the Resource Steps panel when you register this device are:

turn on/off high-voltage

Parameters: N/A

Description: Turns the detector high-voltage on or off

start acquire

Parameters: N/A

Description: Starts data acquisition with a sampling period of **request period** (in seconds) or the selected detector Integration Time, whichever is longest. The Sequence continues on immediately to the next Step after this Step has been issued. Frequently this will be a wait Step.

stop acquire

Parameters: N/A

Var Name: category name (optional)

Description: Ends the data acquisition and associates the resulting profile plot with the specified category. If the current sample (based on sample index) already has a category by this name or if the sample name has changed, then the sample index is automatically updated and a new sample listing is created. If no category is specified, the sample index is automatically updated, a new sample listing created, and the profile plot is associated with the default "sample" category. This profile plot is now the **active plot** upon which subsequent plot commands will operate. The last five profile plots of any category are automatically stored in memory. The **save data to file** command must be issued to save a profile plot to disk.

5.2.4 USB DAQ (Measurement Computing)

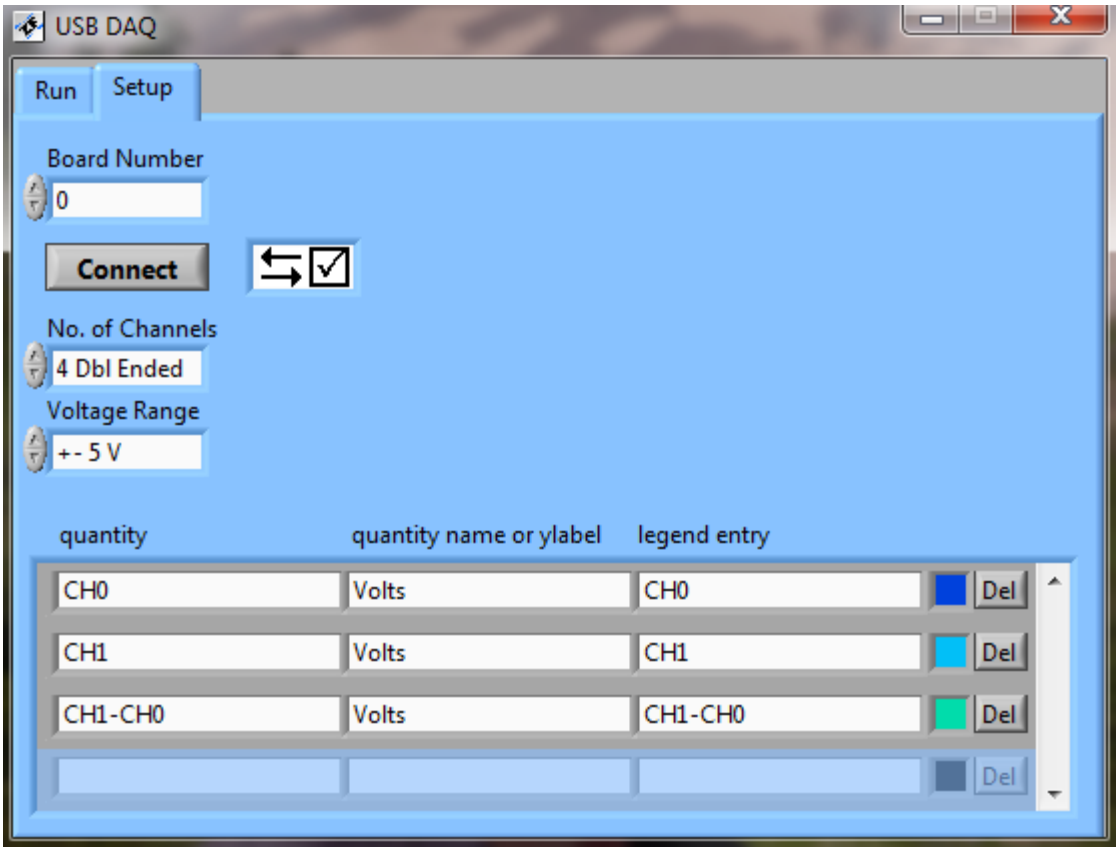
Communication with the Measurement Computing DAQ card is performed via USB. Before the device can be operated from within FloZF, the following setup steps must be taken.

1) Install the device driver, InstaCal, from the CD that came with the USB DAQ and follow the installation instructions.

2) Plug in the USB cable to the USB DAQ
3) Plug in the Cable to the computer. The computer should recognize the device if InstaCal has been installed.
4) Launch the InstaCal application. Two windows appear. The Plug and Play Board Detection Window displays "The following plug and play devices have been detected". In the box bellow the message is listing for the particular USB DAQ including the serial number.
5) Click the **OK** button in the **Plug and Play Board Detection** window. This window closes and the device information shows up in the main InstaCal window. Take note of the board number appearing in the listing for this new device. You will need the board number in FloZF.

Note! The USB DAQ must be connected to the computer before starting the software.

Communication setup for a USB DAQ (Measurement Computing) device is performed from the Setup page of the Instrument Panel.



USB DAQ (Measurement Computing) Instrument Panel: Setup page

The **Board Number** obtained from InstaCal is the only communication setting that must be applied. Communication is tested by pressing **Connect** on the Setup page of the Instrument Panel, and successful communication produces the ⇆☑ symbol.

The USB DAQ (Measurement Computing) can be operated in either a 4-channel differential or a 8-channel single ended configuration. To use a particular configuration, select the corresponding **No. of Channels** in the software.

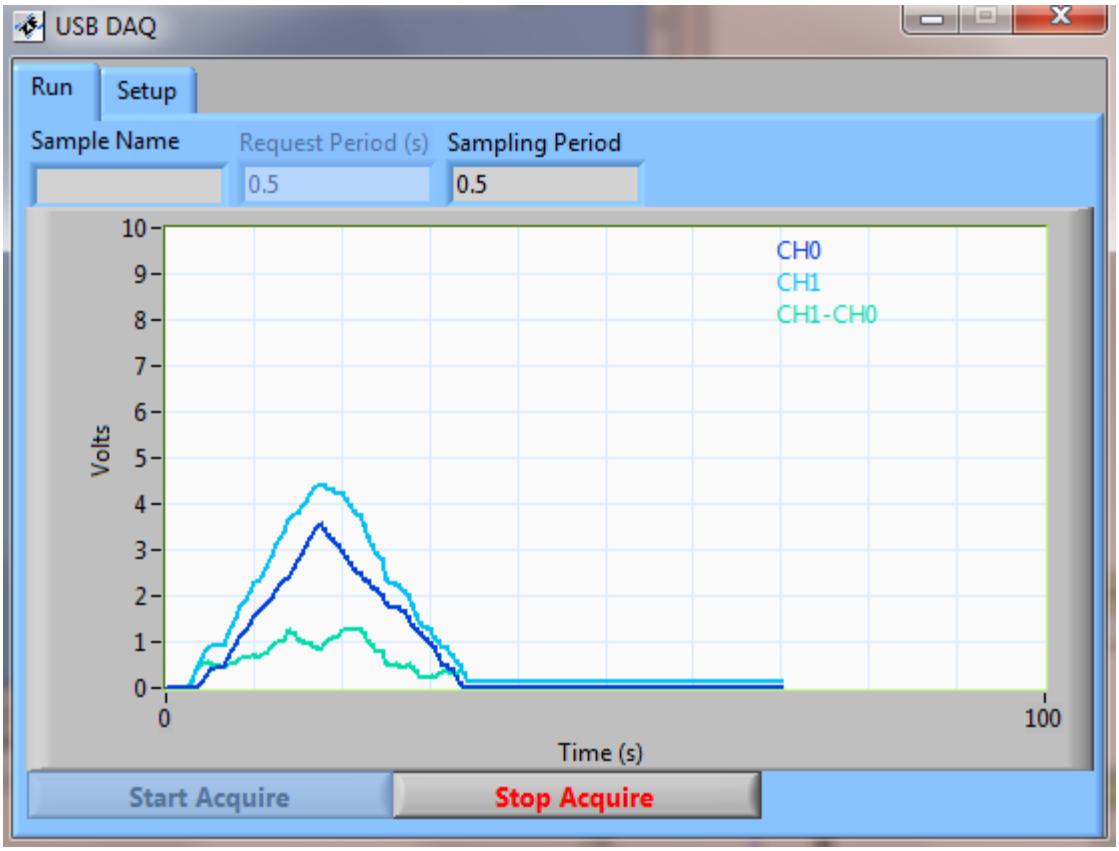
For the differential mode, the **Voltage Range** can be selected as either +-20 V, +-10 V, +-5 V, +-4 V, +-2.5 V, +-2 V, +-1.25 V, or +-1 V. For the single ended configuration, this value is fixed at +-10 V.

At the bottom of the setup page is a table for specifying what combination of channel results to monitor over time and how to display the results. The first column, **quantity**, is used to indicate what combination of channel results to monitor. In this column, the is indicated with a prefix of **CH** followed by the number in the range 0-3 for differential and 0-7 for single ended.

Data can be acquired from the Run page of the Instrument Panel if a Sequence is not currently being executed. Data acquisition is initiated and terminated with the **Start Acquire** and **Stop Acquire** buttons, respectively. The Voltage at any given time is displayed in the **Signal** field (see **Error! Reference source not found.**).

The spectrometer can be operated “manually” from the Run page of the Instrument Panel. Time profiles of the quantities chosen on the Setup page are acquired by pressing the **Start Acquire** button. If only a single quantity has been chosen on the setup page, then the current value is displayed in the **Signal** field as the signal is acquired. A history of the signals is also displayed for this case if the **Display Overlay** option is chosen. The history can be cleared by right-clicking and selecting “clear history”.

Data acquired directly from the Instrument Panel (rather than in a Sequence) cannot be saved.



USB DAQ (Measurement Computing) Instrument Panel: Run page

The Steps that are generated for the USB DAQ (Measurement Computing) in the Resource Steps panel when you register this device are:

start acquire

Parameters: N/A

Description: Starts data acquisition with a sampling period of **request period** (in seconds) or the selected detector Integration Time, whichever is longest. The Sequence continues on immediately to the next Step after this Step has been issued. Frequently this will be a wait Step.

stop acquire

Parameters: N/A

Var Name: category name (optional)

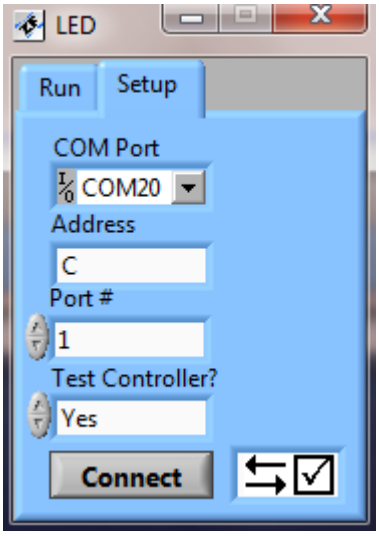
Description: Ends the data acquisition and associates the resulting profile plot with the specified category. If the current sample (based on sample index) already has a category by this name or if the sample name has changed, then the sample index is automatically updated and a new sample listing is

created. If no category is specified, the sample index is automatically updated, a new sample listing created, and the profile plot is associated with the default "sample" category. This profile plot is now the **active plot** upon which subsequent plot commands will operate. The last five profile plots of any category are automatically stored in memory. The **save data to file** command must be issued to save a profile plot to disk.

5.3 Digital IO

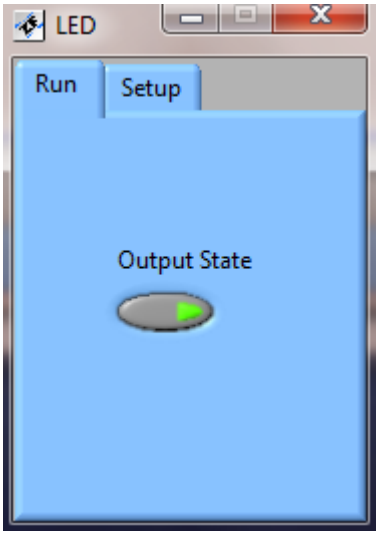
5.3.1 Digital output (MicroLynx and MDrive)

Digital outputs are available on both the MicroLynx and the MDrive controller. Communication settings for the Digital output (MicroLynx and MDrive) are entered on the Setup page of the Instrument Panel. The communication settings to be entered are the **COM Port** and the **Port #**. On the MDrive the Port # is in the range 1-4 and on the MicroLynx this Port # corresponds to the Line number which is in the range 21-26. The location of the ports electrical connections can be found in the hardware manuals. If the controller is to be tested for proper communication, then **Test Controller** should be set accordingly.



Digital output (MicroLynx) Instrument Panel: Setup page

When the **Connect** button is pressed, the status of the Digital output is queried. If the communication test is successful or if **Test Controller** is set to No, then ⇆☑ is displayed.



Digital input (MicroLynx) Instrument Panel: Run page

The digital output can be controlled “manually” from the Run page of the Instrument Panel. Pressing the **Output State** button changes the state of the output.

The Steps that are generated for the Digital output (MicroLynx) in the Resource Steps panel when you register this device are:

turn on

Parameters: N/A

Description: Sets the output state to true

turn off

Parameters: N/A

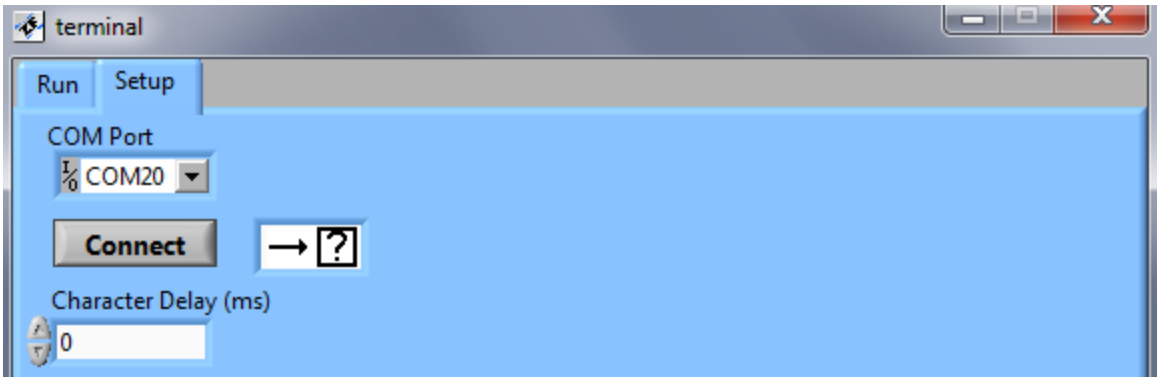
Description: Sets the output state to false

5.4 General Devices

5.4.1 General serial device

The General serial device is intended for sending ASCI-string commands directly to a serial device. It should be used whenever a specific serial device is not already incorporated into the software and provides a versatile facility to users wishing to add new devices not yet supported to their instrument.

Communication setup for a General serial device is performed from the Setup page of the Instrument Panel.



General serial device Instrument Panel: Setup page

Only the **COM Port** is selected directly for establishing communication with a general serial device. When the COM Port is selected, the software automatically attempts to open the port. If there are other registered devices on the COM Port, then the **Baud Rate**, number of **Data Bits**, number of **Stop Bits**, and mode of **Parity** checking are inherited. If there are only General devices on the COM port, these settings can be changed. To change the settings, select the COM Port label in the Registered Devices list (on the Setup page of the main software window) and select the desired settings in the window that appears below. The standard settings are 9600, 8 data bits, 1 stop bit, and no parity checking.

When the **Connect** button is pressed the port is tested to determine if it can be opened with the applied communication settings. If there are no conflicts with devices on the specified COM port, then the communication status indicator changes from ✕ to →[?]. This later symbol indicates that the port is opened to the unknown serial device.

Once the COM port has been opened, the **Character Delay (ms)** can be set. This is the amount of time for which the software should wait between characters when sending messages out on the serial port.

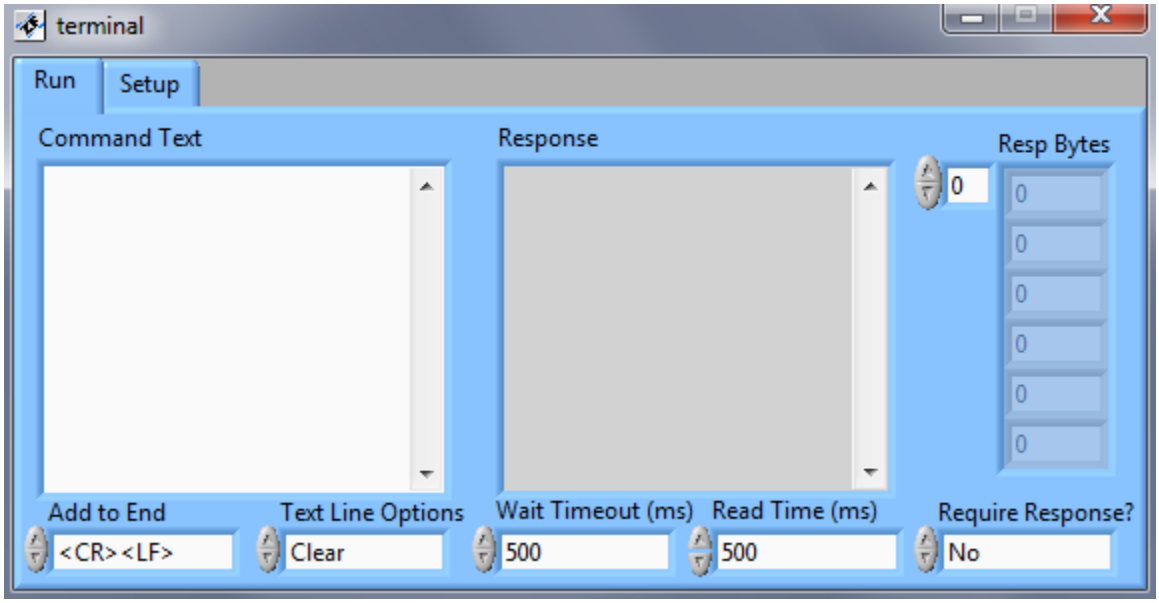
Terminal emulation and the setting of string handling options are performed on the Run page of the Instrument Panel. A field, **Add to End**, allows you to select what characters should be appended to the string before it is sent. Choices are **Nothing**, carriage return **<CR>**, line feed **<LF>**, and both carriage return and line feed **<CR><LF>**.

The **Text Line Options** field allows you to select what should happen to a line of text that is sent to the serial port after it is sent. The choices are to:

- **Clear** the command text that has just been transmitted to allow entry of a new text string,
- **Keep** the text without moving to the next line to allow the same text string to be sent again, and to
- keep the text and move on to a **New** line so that you have a record of what was just sent but the system is ready for you to type in a new command text string.

The last line of the command text field is the line that is transmitted when you press **Enter**.

The **Wait Timeout** is the default time to wait for a response from the device before allowing further communication. The **Read Time** is the default time for reading a response from the device. **Require Response** designates whether or not an error will be flagged if no response is received from the device.



General serial device Instrument Panel: Run page

To use the terminal facility of the general serial device:

1) Choose the correct COM Port and select Connect.
2) Choose what should be added to the end of the string when it is sent. For pumps and valves with MDrive controllers send a **<CR><LF>** for most other devices a **<CR>** will suffice.
3) Choose what to do with the text line after it is sent in the Text Line Options.
4) Type the string to be sent in the **Command Text** box and press **Enter**.
5) Monitor the response in the **Response** box.

The Steps that are generated for the General serial device in the Resource Steps panel when you register this device are:

gen command

Parameters: 'command text', wait time (ms) (optional), read time (ms) (optional), req response (0/1) (optional)

Var Name: response text variable name (optional)

Description: Sends the text **any text** with additions selected in **Add to End** (in Instrument Panel) to the **COM Port**. To send unprintable characters, include their Hexadecimal equivalent inside angle brackets. For example the carriage return character can be sent as <0D>. If **wait time**, **read time**, and **req response** are included then they are used handling a response from the device. Otherwise, the values set in the Instrument Panel are used as described above. The Step is only completed once a response returns from the device, or the **Wait Timeout** time is reached. Any response is stored in the **response text variable**.

prompt for number

Parameters: 'command text', wait time (ms) (optional), read time (ms) (optional), req response (0/1) (optional)

Var Name: returned number variable name

Description: See the description for the general command. The prompt for number command parses the response text for a number and stores the result in the **returned number variable**.

5.5 Mass Flow Controllers

5.6 Plate Samplers

5.6.1 Plate sampler (Nymotion) and Plate sampler servo (Nymotion)

The two plate samplers supported in the software are manufactured by Nymotion. Motion control on the first plate sampler is based on stepper control and on the second plate sampler servo control. Communication settings for the Digital output (MicroLynx and MDrive) are entered on the Setup page of the Instrument Panel. The communication settings to be entered are the **COM Port** and the **Address**.

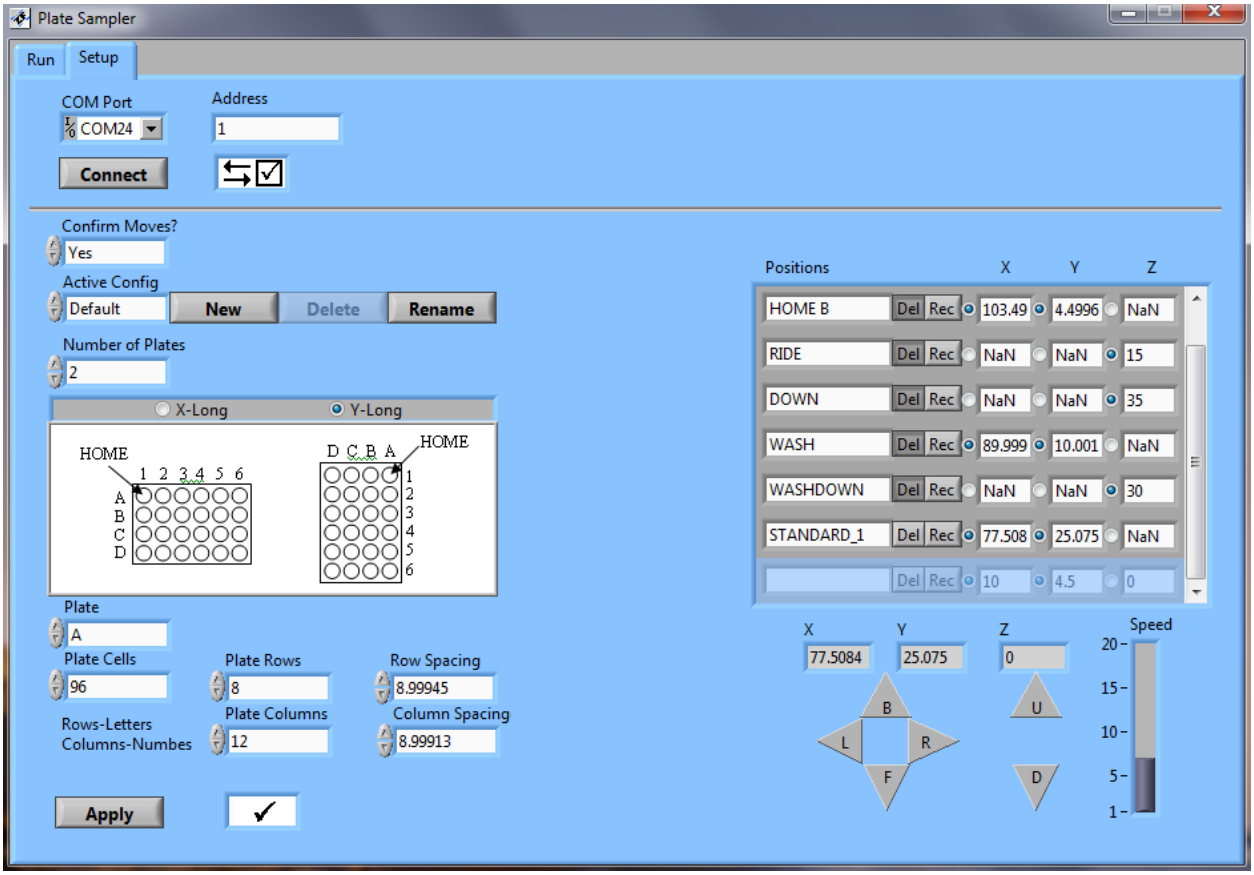


Plate Sampler Instrument Panel: Setup page

When the **Connect** button is pressed, the status of the Digital output is queried. If the communication test is successful, then ⇆☑ is displayed.

Access to plate sampler and plate configuration settings is allowed after communication with the plate sampler has been established. The **Confirm Moves** setting indicates whether the software should prompt the plate sampler for its position after each move. This confirmation step requires additional time and can be set to No if speed is critical. **Active Config** allows the user to select which plate configuration to work with. To create a new plate configuration, select **New** and enter a name for the configuration in the window that appears. Plate configurations can be deleted with the corresponding key. To rename a configuration, select **Rename**, and then enter the new name in the window that appears. **Number of Plates** is used to select whether there are one or two plates in the current active configuration. The orientation for the plates is chosen immediately below with the options **X-Long** and **Y-Long**. Pictures of the orientations are given below the corresponding options. To define the layout of a particular plate, first choose the plate with the **Plate** selector. Then select whether the plate is a standard 96,384, or 1536 plate or if it is a Custom plate. The layout for the standard plates automatically appears in the **Plate Rows**, **Row Spacing**, **Plate Columns**, and **Column Spacing** entries. For a custom plate, measure the plate and enter these values. Numbers on a plate indicate the columns and letters indicate the rows. Additional dimensions must be recorded in the **Positions** on the right-hand side of the window. To record these positions, first choose the **Speed** at which

you want the needle to move and navigate the needle to the desired location using the **R** (**Right** or decreasing X direction), **L** (**Left** or increasing X direction), **B** (**Back** or decreasing Y direction), **F** (**Front** or increasing Y direction), **U** (**Up** or decreasing Z direction), and **D** (**Down** or increasing Z direction) movement keys. Then press **Rec** in the row of the Positions table with the corresponding position name. The default positions are:

HOME A- The X-Y position of the center of well A1 on plate A

HOME B- The X-Y position of the center of well A1 on plate B (if there are two plates)

RIDE- The distance down (Z) to the height at which the needle should travel between plate wells

DOWN- The distance down (Z) to the bottom of the plate wells (assumes all plates are identical depth)

WASH- The X-Y position of the center of the wash station

WASHDOWN- The distance down (Z) to the bottom of the wash station

To define additional positions, type in a name (spaces are not allowed) for the position in a new row of the Positions table and follow the steps described for the default positions. Press the **Del** button to delete a user-defined position. Plate configuration changes are put into effect and saved by selecting the **Apply** button.

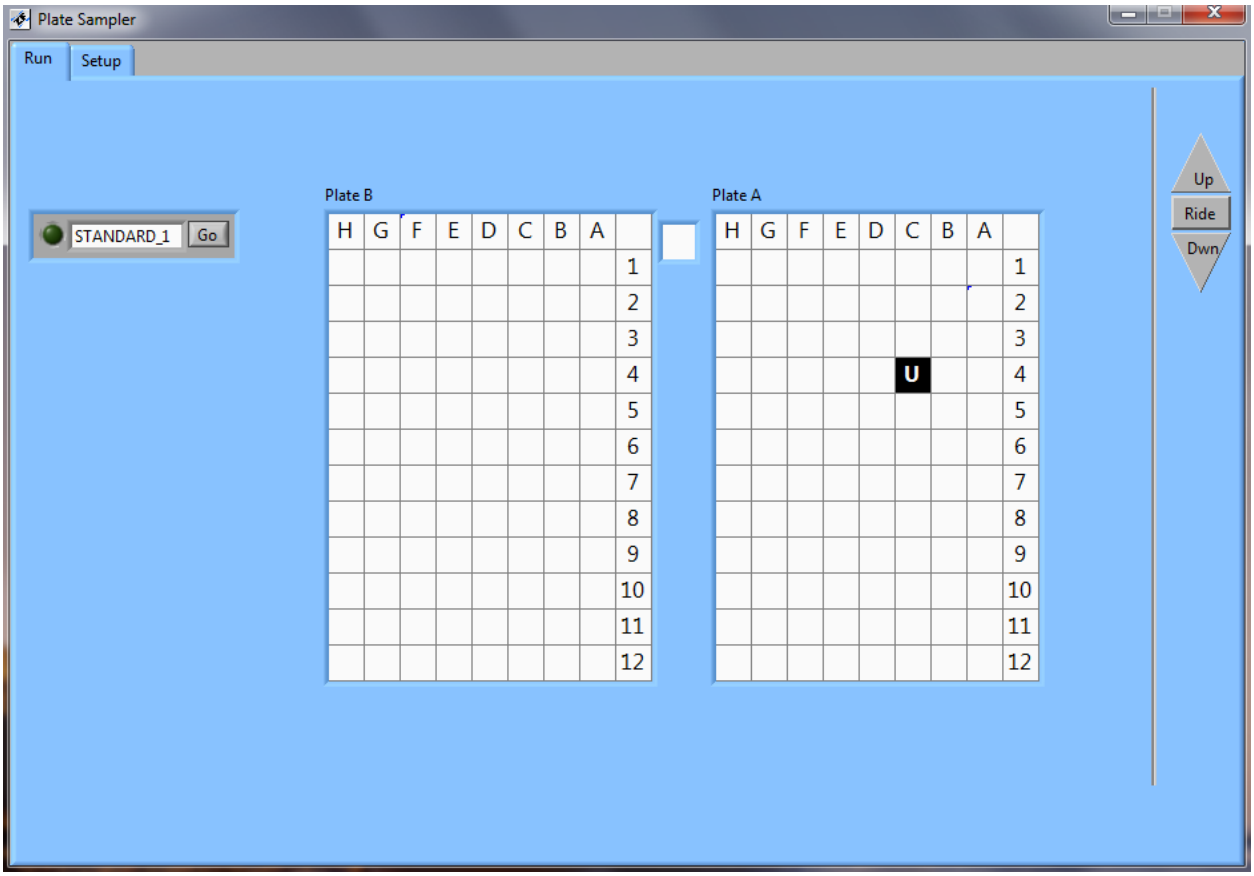


Plate Sampler Instrument Panel: Run page

The plate sampler can be controlled “manually” from the Run page of the Instrument Panel. Representations of the plates are generated on this page automatically based on the plate configuration settings. To move the sampler to a desired plate and well, simply click on the corresponding locations within the appropriate table (labeled **Plate A** or **Plate B**). The white square between the two tables represents the wash station. To move the needle to the up, down or ride height, select the **Up**, **Dwn**, or **Ride** button, respectively, at the far right of the window. The needle at the up and ride heights are indicated with a **U** in the corresponding well position and the needle at the down height is indicated with a **D**. To move the needle to a user-defined position, select the **Go** button in corresponding row in the table at the far left of the window. The indicator for that row will be lit when the needle is at that location.

The Steps that are generated for the Plate Sampler when you register this device are:

go to well

Parameters: well #, plate (A or B)

Description: Moves the needle to the ride height and then to the x-y position of the specified well on the specified plate. The Sequence waits until this move is complete before moving onto the next Step in the Sequence.

dispense head up

Parameters: N/A

Description: Moves the needle to the up position. The Sequence waits until this move is complete before moving onto the next Step in the Sequence.

dispense head down

Parameters: N/A

Description: Moves the needle to the down position. The Sequence waits until this move is complete before moving onto the next Step in the Sequence.

direct to well bottom

Parameters: well #, plate (A or B), hold all (0/1)

Description: Moves the needle to the ride height, then to the x-y position of the specified well on the specified plate, and then immediately to the down position. A value for the **hold all** parameter equal to 1 means that the Sequence waits until the pump has finished before moving onto the next Step in the Sequence. A value for the **hold all** parameter of 0 means that the Sequence continues onto the next Step immediately after telling the plate sampler to move.

adv to next well

Parameters: N/A

Description: Moves the needle to the ride height, then to the x-y position of the next well on the current plate. If the needle is already at the last well on the plate, then it returns to the first well. The Sequence waits until this move is complete before moving onto the next Step in the Sequence.

go to wash station

Parameters: N/A

Description: Moves the needle to the ride height, then to the x-y position of the wash station. The Sequence waits until this move is complete before moving onto the next Step in the Sequence.

down into wash

Parameters: N/A

Description: Moves the needle to WASHDOWN position. The Sequence waits until this move is complete before moving onto the next Step in the Sequence.

direct to wash bottom

Parameters: hold all (0/1)

Description: Moves the needle to the ride height, then to the x-y position of the wash station, and then immediately to the down position. A value for the **hold all** parameter equal to 1 means that the Sequence waits until the pump has finished before moving onto the next Step in the Sequence. A value for the **hold all** parameter of 0 means that the Sequence continues onto the next Step immediately after telling the plate sampler to move.

go to user position

Parameters: 'position name'

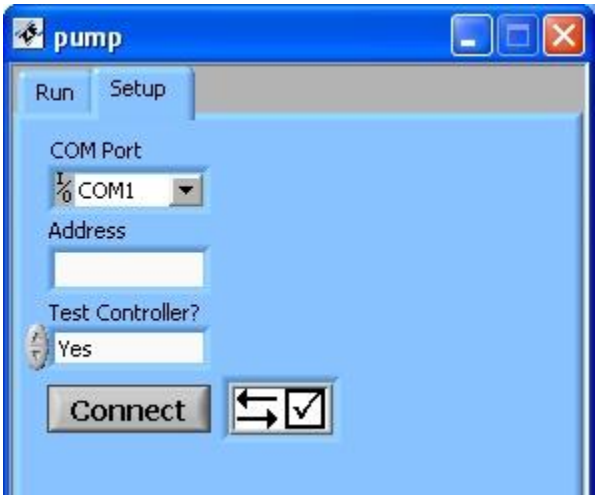
Description: Moves the needle to the up position and then to the x-y-z position of the named position. The Sequence waits until this move is complete before moving onto the next Step in the Sequence.

5.7 Pumps

5.7.1 milliGAT and SmoothFlow pumps

The milliGAT is available with the MicroLynx or RMS external controllers or with the MDrive integrated controller. This distinction is important when selecting a device from the Available Devices List.

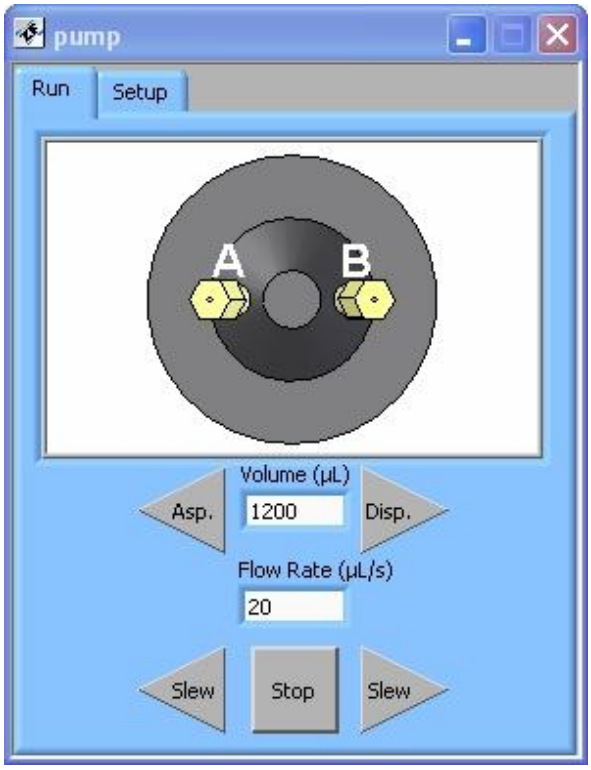
Communication settings for the milliGAT are entered on the Setup page of the Instrument Panel below. Note that the RMS controller communicatesat 57,600 baud and has two stop bits as well as the standard eight data bits and, no parity checking or flow control.



milliGAT Instrument Panel: Setup page

The communication settings to be entered are the **COM Port** and the **Address**. When the COM Port is selected, the software automatically attempts to open the port. Addresses are typically upper case letters such as A,B,C….. If there is no Address assigned to the controller, then this field should be left blank. If the controller is to be tested for proper communication, then **Test Controller** should be set accordingly.

When the **Connect** button is pressed, the pump echo mode is turned on and the status of the pump is queried. If the communication test is successful or if **Test Controller** is set to No, then ⇆☑ is displayed



milliGAT Instrument Panel: Run page

The pump can be operated "manually" from the Run page of the Instrument Panel. When the **Asp**. (Aspirate) or **Disp**. (Dispense) button is pressed, the fluid is pumped in the indicated directions at the rate entered in **Flow Rate (µL/s)** until the volume entered in **Volume (µL)** has been reached. When either of the **Slew** buttons is pressed, the fluid is pumped in the indicated directions at the rate entered in **Flow Rate (µL)**. All of the pumping actions are halted by pressing the **Stop** button. The volume of fluid that has been pumped during the any action is displayed in the field **Volume (µL)**.

The Steps that are generated for the milliGAT (MicroLynx) in the Resource Steps panel when you register this device are:

aspirate or dispense

Parameters: volume(µL), flow rate (µL/s), hold all (0/1)

Description: Pumps the **volume** of fluid at the specified **flow rate** in one direction for aspirate and the opposite direction for dispense. Typically the fluidic manifolds are configured such that dispense pumps fluid in the direction of a selection valve. If two-way communication to the pump is working, this command does not tell the pump to start until it has been probed and found to be stopped. A value for the **hold all** parameter equal to 1 means that the Sequence waits until the pump has finished before moving onto the next Step in the Sequence. A value for the **hold all** parameter of 0 means that the Sequence continues onto the next Step immediately after telling the pump to start. Any further commands to that pump are delayed until the pump is finished pumping.

slew

Parameters: (+/-) flow rate (µL/s)

Description: Pumps fluid at the specified flow rate in the direction indicated by the sign of the value. A positive flow rate results in a dispense and a negative flow rate results in an aspirate. This command executes immediately without waiting for the pump to finish its current pumping action.

stop

Parameters: N/A

Description: Stops the pump

home (MDrive only)

Parameters: (+/-) flow rate (µL/s)

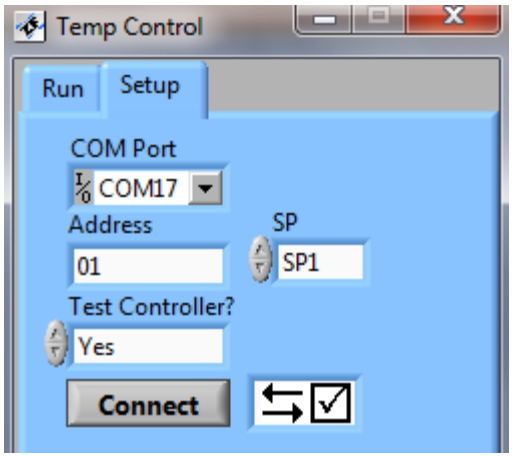
Description: Homes the pump to indexing position at the speed and direction indicated by **flow rate**.

5.8 Software Triggers

5.9 Temperature controls

The software provides communication with temperature controls by both Omron and Newport.

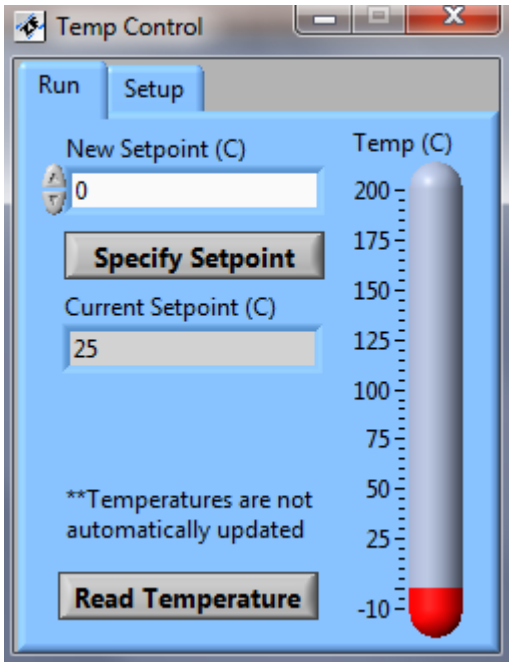
Communication settings for the temperature control are entered on the Setup page of the Instrument Panel. When the COM Port is selected, the software automatically attempts to open the port. The communication settings to be entered are the **COM Port**, the **Unit Number** of the controller. The Newport controller also requires selecting which setpoint, **SP**, to use. If the controller is to be tested for proper communication, then **Test Controller** should be set accordingly.



Temperature control (Omron) Instrument Panel: Setup page

When the **Connect** button is pressed, the status of the valve is queried. If the communication test is successful or if **Test Controller** is set to No, then ⇆☑ is displayed

The temperature control can be operated "manually" from the Run page of the Instrument Panel. A set point is applied by entering the desired value in degrees Celsius in the **New Setpoint (C)** field and pressing the **Specify Setpoint** button. The **Current Setpoint** is indicated in the corresponding field and the current temperature is obtained and displayed on the thermometer by selecting **Read Temperature**.



Temperature control (Omron) Instrument Panel: Run page

The only Step that is generated for the Temperature control (Omron) in the Resource Steps panel when you register this device is:

specify setpoint

Parameters: temperature (C)

Description: Changes the setpoint of the control to **temperature** in degrees Celsius.

wait until steady

Parameters: tolerance (C)

Description: Holds the sequence at this step until the temperature is within the specified **tolerance** of the setpoint.

5.10 Valves

5.10.1 Selection valve

Valco selection valves are equipped with a possibility of three different controllers or actuators. Valco make both a stand-alone actuator and an integrated universal actuator. Both of these actuators are simply indicated by Valco in parentheses. The third controller is the MDrive integrated controller. These distinctions are important when selecting a device from the Available Devices List.

Communication settings for the Selection valve are entered on the Setup page of the Instrument Panel . The communication settings to be entered are the **COM Port** and the **Address**. When the COM Port is selected, the software automatically attempts to open the port. The Address for the Valco controllers should include the "/" symbol for the MDrive, just the upper case letter Address should be entered. If there is no Address assigned to the valve, then this field should simply be left blank. If the valve controller is to be tested for proper communication, then **Test Controller** should be set accordingly.

When the **Connect** button is pressed, the status of the valve is queried. If the communication test is successful or if **Test Controller** is set to No, then ⇆☑ is displayed.

Selection valve Instrument Panel: Setup page

The **Number of Ports** is automatically acquired from the controller if two-way communication is established with the valve. Otherwise, the number of ports needs to be entered in this field.

The valve can be operated “manually” from the Run page of the Instrument Panel shown below. Clicking near a number on the dial sends the valve to the associated port.

Selection valve Instrument Panel: Run page

The Steps that are generated for the Selection valve in the Resource Steps panel when you register this device are:

go to port

Parameters: port #

Description: Sends the valve to the specified **port #**. Often it is convenient to make copies of this step, changing the port # and step name to correspond to what is plumbed to the port.

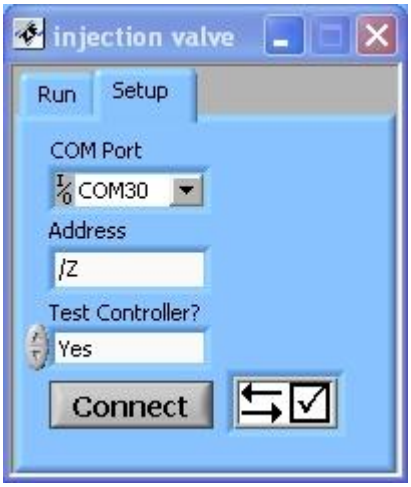
Initialize (MDrive only)

Parameters: N/A

Description: Locates the valve home position for proper alignment with the ports.

5.10.2 Injection valve (Valco)

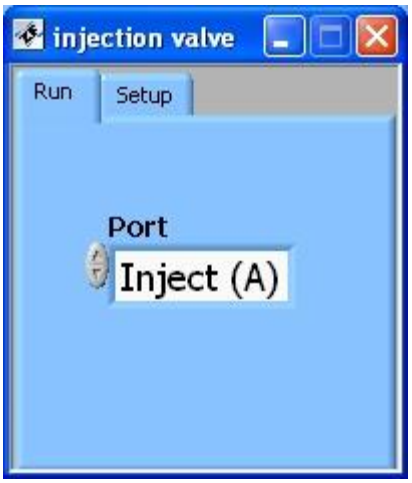
Communication settings for the Injection valve (Valco) are entered on the Setup page of the Instrument Panel shown below. The communication settings to be entered are the **COM Port** and the **Address**. If there is no Address assigned to the valve, then this field should simply be left blank. If the valve controller is to be tested for proper communication, then **Test Controller** should be set accordingly.



Injection valve (Valco) Instrument Panel: Setup page

When the **Connect** button is pressed, the status of the valve is queried. If the communication test is successful or if **Test Controller** is set to No, then ⇆☑ is displayed.

The valve can be operated "manually" from the Run page of the Instrument Panel. Switching of the valve is initiated by selecting the desired **Port**.



Injection valve (Valco) Instrument Panel: Run page

The Steps that are generated for the Injection valve (Valco) in the Resource Steps panel when you register this device are:

inject

Parameters: N/A

Description: Sends the valve to port A.

load

Parameters: N/A

Description: Sends the valve to port B.